Emerging Pattern Mining to Characterize Language Registers in French

Ines Dabbebi

Université de Tunis

Institut Supérieur de Gestion

MASTER RECHERCHE

Spécialité

Sciences et Techniques de l'Informatique Décisionnelle

OPTION

Intelligence et Gestion de Connaissances

Emerging Pattern Mining to Characterize Language Registers in French

Inès DABBEBI

Nicolas BÉCHET	Maitre de Conférence, IRISA-FRANCE	Directuer de Mémoire
Gwénolé LECORVÉ	Maitre de Conférence, IRISA-FRANCE	Co-directeur
Lilia REJEB	Maitre Assistant, ISG-TUNIS	Superviseur

Laboratoire/Unité de recherche: IRISA UMR 6074-FRANCE

Acknowledgments

Foremost, I would like to express my sincere gratitude to my advisors Dr.Nicolas BÉCHET, Dr.Gwénolé LECORVÉ and Dr.Lilia REJEB for the continuous support of my master thesis and research, for their patience, motivation, enthusiasm, and immense knowledge. Their guidance helped me in all the time of research and writing of this master thesis.

I take this opportunity to express gratitude to EXPRESSION team members and to all of the IRISA laboratory members for their help and support during my stay in France.

My last thanks and respects to return to my family. To my brothers who have always supported me and finally to my parents for the unceasing encouragement, support and attention.

Inés DABBEBI

Contents

List of Figures

List of Tables

List of Algorithms

Introduction

Context

Over the last 3 decades, the interest in the linguistic study of phraseology has grown rapidly, especially the corpus linguistics research field. This operation has attracted more than one area: answering questions, biomedical, structure or ontologies. This has spawned the discipline of information extraction, which aims to extract essential details of a particular domain of textual data. The language register or level of formality of language depends on the wealth of the vocabulary used, the complexity of the syntax, mode and time of verbs (etc.). Three main registers in French could be distinguished: the sustained register (or high or formal), the neutral register (or standard or current), and the familiar register. The language registers, by their nature, are easily identified manually by the identification of markers that characterize these styles. For instance, the formal language is recognizable by its rich vocabulary, the use of rare modes and time as imperfect subjunctive and the complexity of its syntax that is opposed to the familiar register, which is characterized by the use of familiar or slang terms, a simple or incorrect syntax and the contraction of certain terms such as discordant.

However, the automatic extraction of the characteristics of language registers is considered as a challenge since it requires the development of methods work on the extraction of information that may add value to users. The data mining methods like association rules, supervised and unsupervised classification, offer different solutions to research problems . Therefore, the main idea of these methods is to discover frequent patterns, which are a set of items or subsequences that occurs frequently in a data set such as finding inherent regularities in data, or to extract emerging patterns, which are patterns that support changes significantly from one dataset to another in order to highlight the contrast and the dissimilarity between different textual contents (text corpora). Despite the size of text corpus, they are considered as difficult problems that require effective methods to cover the search space that may be large to analyse it.

Our main purpose is to explore linguistic patterns present in texts such as the negation without *«ne »* or the apocope of the pronoun *«tu »* data with minimal human intervention in order to discover contrast and specificities of each language register in French.

Motivation

The powerful boost of linguistic patterns in the sociolinguistic and phraseology fields encouraged the development of methods to obtain these patterns. However, regarding to the algorithmic complexity, most methods are limited to only extract frequent patterns (Agrawal et al., 1994) or their representation (Pasquier et al., 1999). This quest for speed has hidden a key objective, namely the discovery of new and significant information based on other properties. The mass of frequent patterns that may be too long, cannot be exploited directly and cannot all be relevant and suitable for the user.

Moreover, the use of frequent patterns is limited. It does not allow, for example, to find exceptions or contrasts between several classes like the emerging pattern. The discovery of emerging patterns under constraints is intended to select the different patterns that satisfy a predicate specified by the user, called constraint and at the same time discover the difference between various text contents. For several years, the literature has abounded many works to dedicate one or more specific constraints (Srikant et al., 1997). Several studies have also proposed to collect constraints that share the same formal properties (Toivonen 1997, Ng et al., 1998, Pei et al., 2001). It is then possible to use a dedicated algorithm in different classes in order to extract emerging patterns.

Contribution

Our ongoing work on one hand, is based on the consideration of the linguistic aspect of French and on the other hand on the combination of different data mining methods. The ultimate goal is to move towards a method to guide the discovery of knowledge from texts by combining data mining methods. The interest in mining methods based on emerging patterns and the desire to automatically acquire patterns that represent the characteristics at lower cost by selecting the most relevant information that represent the contrast between the different language register.

Outline

Our ongoing work is divided into two parts. The first part is devoted to the extraction problem of textual data characteristics while presenting different existing approaches during chapter 1. After that, the second chapter will focus on different methods of pattern mining while focusing on their strengths and their limitations.

The second part, specifically in chapter 3, will introduce the used approach in this work while evoking the basic notions. Furthermore, chapter 4 will present the obtained results, accompanied by a discussion following an evaluation.

Finally, the last chapter will be subject to possible conclusion and perspectives that can improve our approach.

Part I
Theoretical Aspects

Part I presents the theoritical aspects of this master thesis. It provides the necessary background regarding the basic concepts of the sequential pattern mining approaches and the extraction of emerging patterns under multiple constraints approach that represent the core of this master thesis.

Chapter 1

Characterization of Language Registers

1.1 Introduction

The study of expressiveness or phraseology including stylistics, is a research field that has been investigated over the past 30 years by the linguistic community. Due to the explosion of available textual data, the need for efficient processing of texts such as the identification of language register has become crucial to many applications. Nevertheless, there is not an existing method that deals with the extraction of language register characteristics that help to identify different language registers. However, there are methods for the characterization and the identification of the style and the author of textual data known as the identification of authorship. The identification of authorship falls into the category of text classification, an interesting sub-field of text categorization. This field deals with the properties of the form of linguistic expression of the content of a text. Actually, various feature sets and methods have been proposed in the literature. In this chapter, we will introduce the language register and we will present the existing methods for identifying textual resources.

1.2 Language Register

The sociolinguists is a domain that describes several possible relationships between language and society. It aims to explain the variety of language register and to highlight the difference between them based on the impact of linguistic structure or behavior, such as the age-grading phenomenon whereby young kids talk in a different way from the older kids and the impact of society. The major idea of linguistic register has been described by trudgill(1983) as linguistic varieties linked

several criteria starting with occupations, professions or topics have been describing the different registers. The register of law, for example, is different from the register of medicine, which in turn is different from the language of engineering and so on. Registers are usually characterized solely as vocabulary differences; either by the use of particular words, or by the use of words in a particular sense.
there is another definition depicts registers as a special case of a specific sort of language being produced by the social situation.
Moreover, Halliday, Mcintosh and Strevens (1964), they also refer to register as distinguished by utilization. In sociolinguistics, registers represent a combination of languages that relates to communication situations. For instance, talking in a formal setting is different to an informal one. The idea of language register is based on the linguistic practices of the human being, which it could be judged positively or negatively. Undoubtedly, there are a set of registers that make french a rich and fluctuated language. There are numerous registers that could be identified, however, there are no reasonable limits between them that makes register classification a complex problem. Indeed, even a language that appears to have a simply social, non-educational capacity, it may express the conformity to a group norm, still discuss the basic problem «I do/don't belong to the same group as you», and tries to express that as clearly as possible.

1.2.1 Degree of Linguistic Formality

The degree of formality is represented in language register through different variables, which expresses social meaning, for example, in french, the omission of the negative particle «ne».
These variables do not change the meaning of the textual content, but they carry various sorts of information such as information about the person, the recipient, the situation, and the context.
French sociolinguistic and didactic researchers, on social and situational variation, have identified linguistic features, which connote a certain degree of formality.
The socio-style of french has been analysed by sociologists in terms of various scales.
For instance, Cadet (2003) identifies four levels of formality: formal, standard, informal, and colloquial.
Likewise Arrivé, Gadet, and Galmiche (1986) propose also a four-level scale: formal, every day/medium, informal, and colloquial.
In return, Mougeon et al. (2000), present a tripartite division: vernacular, mildly marked, and formal. These levels are defined in terms of negativity and social and situational variation. A classic problem with sociolinguistic field is the measurement of variation between different genres or registers. The stylistic variation results from the fact that different people express themselves in different ways, and that the same person may express the same idea quite differently when addressing different audiences, using different modalities, or tackling different tasks. However, there is no a clear or general definition of «formality».
The following subsections focus to introduce the three main language register in French: Formal, neutral and familiar registers.

1.2.1.1 Formal register:

This type of language is often learned and repeated by rote. We can find it in writing instructions, such as public notices, biblical verse, prayers, the pledge of allegiance, and so forth.
Rey et al (2010) defined the formal register as a type of discourse used in specific situations where the formal speaker is exceptionally careful about the articulation and choice of words and sentence structures, which they try to approximate as closely as possible to the standard form of rule of the language, it is considered as a typical careful speech and written french that is emphatically connected with individuals of the upper social strata. This is the register used in the most academic and scientific publishing or debates and ceremonies. This style is considered as impersonal and frequently takes after a prescriptive format. It can be recognized in the use of a richer vocabulary, more complex sentence structures, figures out more elaborate style and usage patterns and verb tenses that are normally little used. The speaker avoids slang and may use technical or academic vocabulary. It is likely that the speaker uses less contractions, yet pick rather for complete words. In this situation, an author or speaker is more likely to use vocabulary with latin or greek roots. As instance, in a scientific article, the writer may be more likely to use the word in latin root «une dame»than «une femme».
In another situation like consulting an expert, for example, the speaker is likely to address the expert by a title such as *«docteur», «Mr.«Mme.»*.
This register is rarely used in spoken but highly used in novels.
For example, *je n'ai point lu cet ouvrage* where the use of the word «point»instead of «pas»is considered as a rare word.

1.2.1.2 Neutral register:

The neutral register is considered as a median or unmarked language, bearing no obvious colouring.
It is also represented as a general or common register that is linguistically neutral (eg. Morphology is common to all speakers natives). It is used both in writing and orally in everyday contexts.
For instance, *je n'ai pas lu ce livre*, this sentence respect the French's grammatical rule.

1.2.1.3 Familiar register:

Labère (2004) represented the familiar register as conversational in tone. It is the language used among and between friends and members of the family to communicate between each others. Words are more general, rather than technical. This register may include more slang and colloquialisms.
At this register, speakers are more likely to use vocabulary words with an anglo saxon or germanic root.
Additionally, the register includes the popular vocabulary, which is used among some specific social groups such as adolescents. In fact, this register does not conform usually to the rules of standard french. However texts in this register, are typical of informal speech, are inappropriate in

formal settings, are associated with speakers from the lowest social level that do not adjust for the most part to standard french and are typical of the informal register.

Some linguists have tried to determine the formality level of a speech by considering the frequency of words and grammatical forms that are viewed as either «familiar»or «careful», such as *«vous»vs. «tu»* or the omission of the negative particle in sentence negations in french.
For example, *ce bouquin, je l'ai pas lu, moi*, in this sentence there is characterised by the omission of «ne»and the use of an anastrophe phrase.

1.3 Literature Reviews

Textual data, to be more specific language register in French, have not a specific method in order to identify the characteristics of each register. However, there are different approaches that aim to identify text and extract text specificity based on the authorship analysis approach or the natural language processing approaches.

1.3.1 Authorship of Textual Resources Approach

In the text mining area and computational linguistics domain, there are three classical classification issues: topic classification, genre classification, and authorship classification. The most difficult problem among the three classification issues is experienced while classifying textual corpus in terms of their authorship called also authorship classification, authorship attribution or authorship discrimination. This issue can be considered as classifying textual corpus problem based on the writing styles of the authors. It is a nontrivial issue notwithstanding for the human being: while people can undoubtedly distinguish the subject and the type of texts, the act of recognizing and identifying the authorship of texts is harder. Even worse, if texts have the same subject and type, the task turns out to be much harder.
This section presents a brief review in authorship analysis based on a survey of writing-style characteristics, analytical techniques and other related parameters.

1.3.1.1 Authorship Analysis

The main idea about statistical or computer-supported authorship attribution is that by measuring some textual features we can distinguish between texts written in different styles.

The authorship analysis as illustrated in figure 1.1 can be defined as a process that attempts to examine different features of a given text in order to gather interesting information on its authorship. The main concept of this process is based on a linguistic research field known as stylometry that refers to statistical analysis of literary style. Gray, Sallis, and MacDonell (1997) distinguished three major fields in authorship analysis studies.

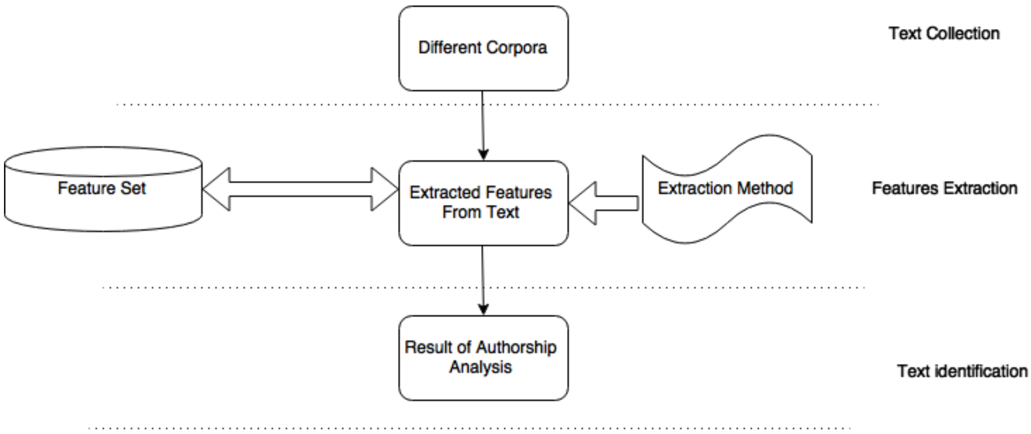

Figure 1.1: Authorship analysis process.

Firstly, **the authorship identification** determines the likelihood of the text corpus to be created with a specific style by looking at different works of the author. In some literatures, it is also known as authorship attribution.

Secondly, **the authorship characterization** summarizes the attributes of an author and creates the author profile in lights writings of the author. Some of these characteristics incorporate gender, educational and social background, and the nature dialect.

Thirdly, **the similarity detection** looks at numerous text corpora in order to compare them after that it attempts to figure out if they were created by the same author without really recognizing the author's profile.

Given a set of texts of different authors that have different styles. The attribution of a new text to one of the author is considered as a statistical hypothesis test or a classification problem. The core of this classification is to distinguish a set of characteristics that remain moderately consistent with a huge number of works made with the same style. When a list of features has been specified, a given text writer can be represented by a $n-$dimensional vector, where n is the total number of features. Hence, the set of features and the analytical methods essentially have an impact on the performance of the authorship identification.

The following section focuses on features selection that consider as a core of authorship analysis, extraction steps and will present the related literatures.

1.3.1.2 Stylometric Features

The study on text representation features for stylistic purposes is principally centered around the computational requirements for measuring them. Zheng et al (2006) present three main features: Lexical, syntactic and structural features. First and foremost, lexical characteristics consider correspondingly a text corpus as an abstract set of word-tokens or characters. However, syntactic features are more complex than character features. At that point, syntactic features require more profounder linguistic analysis, while structural features must be characterized only in certain text domains or languages.

Lexical Features

Zheng et al (2006) present straightforward and simple method to view a textual content is an arrangement of tokens assembled into sentences, where every token is compared to a word, number, or even a punctuation or accentuation mark. The basic lexical feature is shown in the table 1.1. The primary attempts to attribute authorship were based on simple measures, for example, sentence length counts and also word length counts. In certain textual content areas with overwhelming utilization of shortenings or acronyms like in an email or short messages, the lexical feature selection may present significant noise in the measures, for instance, the use of the acronym *«HT»* may correspond to the verb *«acheter»* or *«Hors Taxes»* in French. The vocabulary richness functions are attempting to measure the differing qualities of the vocabulary of a textual content. The dominant part of authorship attribution studies is (at least partially) taking into account lexical characteristics to represent the style. The text content is considered as a bag-of-words, which is a sparse vector of occurrence counts of words, every word having a frequency of occurrence without taking into account contextual information. The selection of particular words that will be used as features is typically based on arbitrary criteria that require language-dependent expertise.

	Features
lexical	Token based: Word length and sentence length
	The word itself
	Word frequencies
	Vocabulary richness

Table 1.1: Lexical Features.

A straightforward and exceptionally successful method characterizing a lexical feature list for authorship attribution is to concentrate and to extract the most frequent words found in the accessible corpus. At that point, a choice must be made about the measure of the regular words that will be used as features. Madigan, et al. (2005), used all the words that occur at least twice in the corpus.

This methodology gives a basic and effective arrangement of words, however it ignores word order information like contextual information. For instance, the expressions «take on», «the second take«take a bathsimply give three meanings of the word «take».

Syntactic Features

A more expand content representation approach is to use syntactic information. The major idea is that authors have a tendency to employ unconsciously comparative syntactic patterns. Thus, compare to lexical information or syntactic information, it is considered a more reliable authorial fingerprint. In addition, the success of lexical features in describing the document style presents the usefulness of syntactic information since they normally experienced in certain syntac-

tic structures. Moreover, the syntactic measure extraction is considered as a language-dependent methodology since it depends on the accessibility of a parser that have the capability to evaluate a specific common natural language with relatively high accuracy. Baayen et al. (1996), introduced the usage of syntactic features in order to achieve authorship attribution. In light of a syntactically annotated english corpus, they identify,extract and rewrite rules. Every extracted rule represents a part of syntactic analysis. For example, the following syntactic rule
A $\mapsto P : PREP + PC : NP$ present an adverbial prepositional expression (A).
This rule is constituted by a preposition word (PREP) after that a noun phrase (NP), which constitute a prepositional complement (PC). This information presented both what the syntactic class of each word is and how the words are combined to form phrases or other structures. This is another example, the following sentence would be analyzed as follows: NP [another attempt] VP [to exploit] NP [syntactic information] VP [was proposed] PP [by stamatatos] where np, vp, and pp refer separately to a noun phrase, a verb phrase and a prepositional phrase. The extracted phrase represented noun phrase counts, verb phrase counts, length of noun phrases, length of verb phrases, etc. The possible syntactic features are presented in the table 1.2.

Features	
Syntactic	Sentence and phrase structure
	word structure
	Rules frequencies

Table 1.2: Syntactic features.

Structural Features

Generally, structural features describe the way a writer or speaker organizes the layout of a textual content. De vel (2000) presented a few basic components (see table 1.3). It describes also a synchronic or diachronic analysis of language on the basis of their structure as reflected by irreducible units of phonological, morphological, and semantic features.

features	
structural	number of sentences
	number of lines
	number of paragraphs
	number of words

Table 1.3: Structural features.

Feature Selection

The primary objective of features selection is a binary decision-in or out-for the inclusion of features in some subsequent analysis. For example, it may help to know, which word is dependably used contrastingly in the different text corpus. This may be for the motivation behind using these features in a subsequent model of individual speaker or for a subjective evaluation of the ideological content, etc. The feature sets combine usually numerous kind of features, which it defines several information that it helps in the sociolinguists study.

1.3.2 Text Characterization Based on Natural Processing Language Approach

The methods based only on the natural language processing (NLP) are among the approaches proposed to the extraction of text specifities. Among these specifities, we can find the characteristic of text documents. However, the methods based on NLP are based on manual drafting of rules (morpho-syntactic patterns) to extract the desired information. Based on this method, Blaschke et al. (1999), was manually generated patterns attempted to extract relationships between proteins. They proposed a model that maps the patterns built on a specific field corpus while using a dictionary of named entities characterizing the names of proteins. These are fixed before starting the extraction procedure. Interactions are not validated by the simple fact of a correspondence with the patterns. The defined patterns are a set of sequences based on morphology and syntax of an expression, thus relying on parsing and labeling part-of-speech (pos-tag). Several patterns are defined in this approach. For instance, [proteins] (NLP) [verbs] (6-10) [proteins], where parenthesis (0-5) means that there may have 0 to 5 terms between [proteins] and [verbs]. The obtained results after experiments show a good performance of this approach, however, disadvantages such as encountered errors when detecting named entities and definitions of all possible patterns make the approach less attractive.

This approach was also used by Zhang et al. (1997) To extract spatial information representatives of geographic features. Their idea is to refer to an annotated corpus to manually extract the syntactic patterns expressing a spatial relation. These patterns are subsequently transformed into rule in order to match the data and to extract the desired information as well. This type of approach can cause errors during the construction of syntactic rules, especially with the long sentences.

These approaches are not much used. Indeed, the extraction of information based on the NLP methods has limitations due to the fact that data are in different formats, which change from one language to another since patterns does not necessarily follow the same morphological and syntactic structures. Thus, to define or to predict every possible combination would be difficult to accomplish, and although the number of rules is optimal, their construction cost both in time and human effort is not negligible. Therefore, this aim the use of other methods based on data mining. This context will be the subject of the next chapter.

1.4 Discussion

Actually, the lack of a good definition of level of formality and the quantification of the dimension of style has hampered sociolinguistic research. It may represent a challenge of textual feature selection problem. Moreover, the high dimensionality of the unique terms like words or phrases, that can contain a little or a huge number of terms, even for a moderate-sized text' collection is considered as a challenge of this problem. The extraction of information from text documents has been the subject of many research fields and several methods have been proposed. However, it is based on manually drawn patterns by studying the syntactic and morphological appearance of sentences and relationships referred for extraction. Although these methods tend to have good accuracy used to extract the information. Indeed, the drawbacks of NLP methods led to the development of more flexible approaches such as methods based on data mining methods.

1.5 Conclusion

In this chapter, we have presented the basic notions of textual features and the level of formality in french. We have also discussed the major approach based on authorship analysis and NLP while presenting their strengths and their weaknesses. In the next chapter, we will present the basic concepts and the major approaches based on data mining in order to extract the characterization of language registers in French.

Chapter 2

Pattern Mining

2.1 Introduction

Natural Language Processing (NLP) and Information Extraction (IE) in particular aim to provide accurate parsing to extract specific knowledge such as textual features by analyzing frequent and emerging patterns. A common feature of IE methods is the need for linguistic resources (grammars or linguistic rules). This chapter presents this problem and the existing methods for discovering patterns using data mining approach. The pattern mining methods aim essentially to discover new information. The core of the mining process is the search of regularity in the data called patterns. For example, from biological situations described by a set of genes, a pattern is considered as a set of genes that are frequently found in many biological situations that may represent a potential biological interest or it may be used to characterize texts (Turmel and al., 2003).

2.2 Frequent Pattern Mining

The data mining area has different distinguishing problems that correspond to classification, clustering and frequent pattern mining.
Compared to other problems, frequent pattern mining has been considered as a focused theme in data mining field for over the last two decades. There are various methods proposed for mining several kinds of patterns, including sequences. Each sequence made of a list of itemsets where each itemset is made of a set of literals called items. Discovering frequent patterns plays a fundamental role in many fields. In fact, abundant studies have been committed to this research, and an important advancement has been made in which a large amount of efficient and scalable algorithms were developed.

The problem of frequent pattern mining has been widely dominated in the literature thanks to

its various applications in diverse domains. It can be used in association rule mining, classification, clustering, and other data mining tasks.
Firstly, frequent pattern mining concept was proposed by Agarwal et al., (1993). The problem was originally proposed in the context of market based analysis in order to extract interesting correlations between products that are bought together.
This concept came into existence where it is needed to discover useful patterns in different transaction databases.
Since then, the interest in the data mining field has increased rapidly along with the wide variety studies in literature.
Despite the level of maturity that has been reached and the important progress in this field that has been made, frequent pattern mining is still considered as one of the most seriously explored problems regarding algorithmic development and much remains to be done.
As a matter of fact, frequent patterns are those items, sequences or substructures that repeat in database transactions with a specified frequency, which is greater than or equivalent to a minimum threshold.

Problem definition

The problem of frequent pattern mining is to discover relationships among the items in a database. We can define it as follows.
Let $I = < i_1, ..., i_n >$ be distinct literals called items. An itemset or pattern X is a non-null subset of I, $X \subseteq I$. An itemset X with size k is called a $k-$itemset.
In frequent pattern mining, the problem is that of finding all patterns P.
Given a transactional database DB with a multi-set of transactions $T = < T_1, T_2, ..., T_N >$, where each $T_i \epsilon T$ and n correspond to the size of the transaction.
-The amount of transaction that contain a given itemset X is called the support of X denoted by $supp(X, DB)$.
-ID is the unique identifier of the transaction, where each transaction represents a database entry.
Pattern mining aims to explore new knowledge from all the extracted patterns. More precisely, pattern mining task is to determine the complete set of itemsets in a given DB with the respect to user requirements.
Hence, the user specifies a constraint in order to get the most efficient patterns according to his point of view and his needs. Agrawal et.al (1993) present the minimal frequency constraint to ensure that the itemsets have a frequency no less than sup_{min} sequence, where sup_{min} is a given threshold set by the user to define a frequent pattern i.e. $sup_{min} > 0$ and Freq(X, DB) $\geq sup_{min}$.

Example

Let DB presented in Table 2.1 be a customer transaction database containing 4 transactions and a given sup_{min} equals to 2. The itemset X=$< a, d, f >$ is a frequent pattern since it is contained in both transaction 2,3 and 4 i.e., supp(a,d,f)=3, which exceeds the sup_{min}.

One of the principle reasons behind the high level of interest in frequent pattern mining algorithms is because of the computational challenge of the task.
Notwithstanding of a reasonable sized data set, the search space of frequent pattern mining may form an exponential number of patterns, which is challenging to find algorithms with better computational efficiency.
Therefore, this causes problems for itemset generation, especially when minimal frequency constraints are low.

$SequenceID$	Items
1	b,c,e
2	a,d,c,f
3	a,d,f
4	a,d,f,e

Table 2.1: A transaction database DB

2.3 Frequent Sequential Pattern Mining

Sequential pattern mining is a data mining technique that intent to discover correlations between events through their order of appearance. It is considered as an important field of data mining with broad applications. It is used in a wide range of applications for different purposes. It can be used in business organizations in order to identify customer shopping behaviors or in web mining to explore several web logs distributed on multiple servers or to study DNA sequences, etc.
However,it is also known as a complex problem regarding the need to examine a combinatorial explosive number of generating sub-sequences patterns.
It is also considered as a significant issue that deals with the discovering a set of attributes, shared across time among a large number of sequences, called pattern and extracting statistically effective patterns between data, which appears sequentially with respect to a specific order.
Given two sequences A= $< a_1, a_2 \ldots, a_n >$ and B=$< b_1, b_2 ..., b_m >$. A is called a sub-sequence of B, denoted as A ⊆ B, if there exist integers $1 \leq j_1 \leq j_2 \leq ... \leq j_n \leq$ m such that $a_1 \subseteq j_1, a_2 \subseteq j_2 \ldots, a_n \subseteq b_{jn}$.
In the same way, B is called then a super sequence of A. For example: Given B=$< a(abc)(acd)(dcf) >$. B is a super sequence of A when A=$< (a)(a)(ac)(dc) >$.
Then, the sequential pattern mining problem can be defined as follows: Consider a given sequence database DB with a set of sequences $< S_1 \ldots S_N >$, as the input, where each sequence S_I consists of a list of customer transactions such that each transaction contains a set of ordered items.
Given a minimum support threshold, further sequential pattern mining is applied to discover all frequently sub-sequences whose occur more than a sup_{min} fraction of the sequences.
An example of sequence database is a sell customer database wherein the sequence or collection of bought products in a specific period of time. A sequential pattern-mining algorithm can be used on this sequence data to extract patterns that are repeated over time and then these extracted patterns can turn in to be used to find associations between the different items or events for different purposes such reorganization, prediction and planning purposes.
Data are represented as a table where each row represents a customer transaction. Such table is called a *«retail customer database »* or an *«Sequence database »*. Let consider a sequence database that contains four sequences $< s_1 ... s_4 >$ and it represents seven different items (a,b,c,d,e,f,g), as described in Table 2.2.
$< (a)(abc)(ac)(d)(cf) >$ is completely different from $< (aab)(c)(a)(cd)(cf) >$.
Given a $sup_{min} = 2$, as a result, 53 frequent sub-sequences can be generated which is considered as a large amount with regards to the number of sequences in the database.

$SequenceID$	Sequence
1	(a)(abc)(ac)(d)(cf)
2	(ad)(c)(bc)(ae)
3	(ef)(ab)(df)(c)(b)
4	(e)(g)(af)(c)(b)(c)

Table 2.2: Retail customer database

Accordingly, sequential pattern mining deals with the problem of scanning a combinatorial large number of frequent sub-sequence patterns.

2.4 Emerging Sequential Pattern Mining

The traditional sequential patterns do not take into account several information like contextual one since patterns extracted from data are usually general. In this section, we introduce another kind of sequential patterns, called emerging patterns. The discovery of powerful contrasts between data sets is an important issue in data mining. Firstly, the idea of emerging patterns was introduced by Dong and Li (1999). Emerging patterns capture the significant changes and differences between sequence database. Emerging patterns are defined as sequential patterns whose support increases significantly from one data set to another one. More particularly, emerging patterns are sequential patterns where their growth rate, which is the ratio of the supports in the two sequence databases, increases significantly until it exceeds a specific threshold.
Accordingly, given a sequential pattern P from a sequence database DB_1 is an emerging pattern to another sequence database DB_1 if Growth-Rate (P) $\geq p$ and Growth-Rate= supp-1 (P) /supp- 2 (P).
The extraction of emerging patterns can basically be regarded as a variation of association rules mining. The itemsets discovered using emerging patterns are considered as significant highlights features that can be used to recognize or describe the distinctions among a collection of sequence databases, while association rule mining discovers rules that only depict the current situation in every sequence databases.
This method was used by Quiniou et al., (2012) for stylistic analysis, from a linguistic point of view, by considering emerging patterns. In fact, their work show that mining emerging patterns of words with gap constraints gives new relevant and generic linguistic patterns. It aims to validate that characteristic linguistic patterns could be identified using data mining techniques.

2.5 Literature Review

Broadly sequential pattern mining algorithms can be classified into different types such as apriori based approaches and Pattern growth algorithms.
These algorithms have further classification and extensions. Detailed explanation of each algorithm with its important features, pseudo code, advantages and disadvantages is given in the subsequent sections of this chapter.

2.5.1 Candidate Generation Approach

Agrawal and Srikant (1994) developed the classical apriori algorithm. This algorithm relies on generate and test approach and an important property: the apriori property. This property is also known as anti-monotone property or downward closure property, and it is a basic pillar of the apriori algorithm. This rule is essentially used in order to reduce the number of patterns combination. It states that all non-empty subsets of a frequent itemset must be frequent. For example, (a,b,c) is a frequent itemset, further all of its subsets $< (a), (b), (c), (ab), (bc) >$ and $< (ac) >$ must be frequent. In the other view, if an itemset is not frequent, then none of its super-sequences could be frequent. As a result, the list of potential frequent itemsets eventually gets smaller as mining progresses.
Otherwise, if A and B are two patterns, B ⊆ A and B is a non frequent pattern, then any pattern A that contains the pattern B is also non frequent. Otherwise, apriori approaches cannot deal correctly with a large sequence database or with a large amount of sequential patterns to be mined.

2.5.1.1 Mining Sequential Patterns by Generalized Sequential Pattern Algorithm

One of the most known apriori-like algorithm is the Generalized Sequential Pattern Mining (GSP). GSP algorithm proposed by Srikant et al., (1995) is based on the generation and the test of the set of frequent sequences known as candidates. The apriori pruning GSP has the ability of scanning the database several times in order to reduce the search space, however, this process is costly. In the first pass, the algorithm starts by detecting all the frequent sequences of length-1 which contains only one item. Then at each pass, GSP scans the sequences of length-k in the database in order to explore all the frequent k-patterns. The frequent $k-$patterns in each level generate the other candidate $k + 1-$sequences. Finally, The algorithm stops when we can not generate another sequence or we can not find any new sequential pattern.
A typical Apriory-like sequential pattern mining method, such as GSP, adopts a multiple-pass and candidate generation-and-test approach is illustrated using the following example:

Example
Consider the sequence database DB (table 2.2), the sequence in S with Sequence-id 1 is listed as $< (a(abc)(ac)d(cf)) >$ instead of $< (a(bac)(ca)d(fc)) >$.

By filtering the rare item g, as illustrated in figure 2.1, we get the initial seed set L1 = $<(a),(b),(c),(d),(e),(f)>$, every part in the set representing a 1-element sequential pattern.
After that, each subsequent pass starts with the seed set of sequential patterns found in the previous pass. This set of sequences is used to generate new hidden patterns, known as candidate sequences, based on the Apriory property.
For L1, a set of 6 length-1 sequential patterns produces a set of 6*6+6*5/2 = 51 candidate sequences,i.e, L2 =<(aa),(ab),...,(af),(ba),(bb),...,(ff), ((ab)), ((ac)),...((ef))>.

Each candidate encloses at each pass one more item than a previous sequential pattern. Then, all candidates with support more than min-support in the database from the set of the newly found sequential patterns. This set is used for the next scan.
The GSP algorithm (see algorithm 1) ends when there is no more new sequential pattern is found in a pass, or when the algorithm stop generating candidate sequences.
Apparently,the amount of scans is in any event the greatest length of sequential patterns.In fact, if the sequential patterns obtained in the last output still produce new candidates,therefore it acquire one more scan.

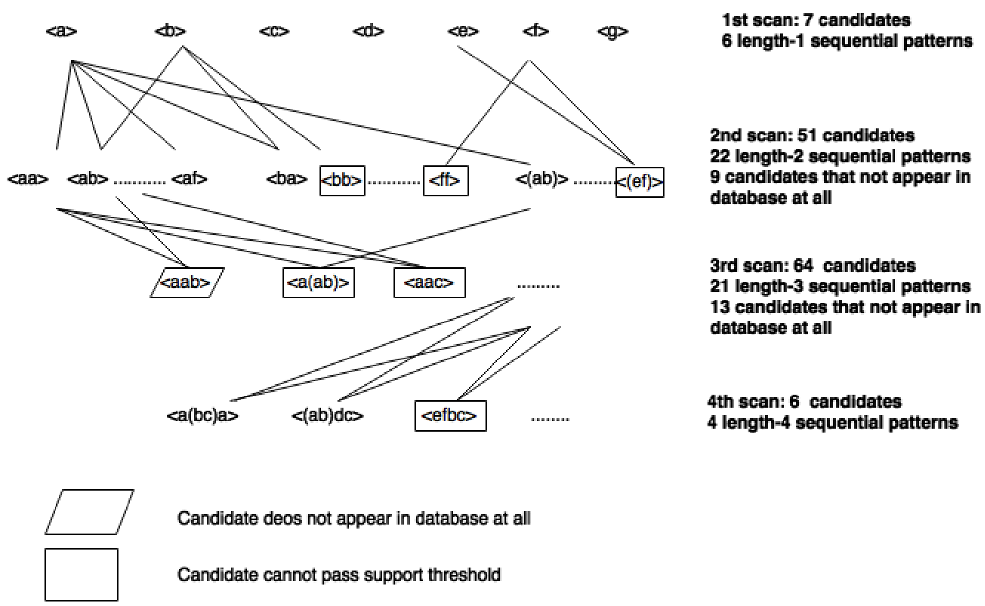

Figure 2.1: Candidates and sequential patterns in GSP (Han et al., 2004).

Algorithm The GSP algorithm is represented as follows:
It takes as input a collection of itemsets of size k, and as an output, it returns only the frequent itemsets of length-1 to length-k.

Algorithm 1 GSP algorithm

Join step *C_K is generated by joining L_{k-1} with itself.*
Prune step *(k-1) not frequent itemset cannot be a subset of a frequent k-itemset.*
Pseudo-Code
Input:
C_K: Candidate itemset of size k
Prune step *L_K: frequent itemset of size k*
Prune step *L: frequent items*
for*(K=1; L_k <>0; k + +)*
C_{k+1} = candidates generated from L_K;
for*(transactions t ϵ T)*
increment the count of all candidates in C_{k+1} that are included in t.
L_{k+1} = candidates in C_{k+1} with minimal support
end for
end for
END
return $\bigcup_k L_k$ =0

2.5.1.2 Bottlenecks of Apropri-Based Approach

The task of finding every single of frequent sequences in expansive databases is considered as a very difficult challenge.
In spite of the advantages of the ariory pruning, according to Pei et al., (2005), the GSP algorithm still develops a large amount of candidates. In the previous example, with a database that contains only 6 length-1 sequential patterns. It generates 51 length-2 candidates, 22 length-2 sequential patterns and 64 length-3 candidates, etc. The problem is that there are some candidates engendered by the GSP algorithm that may not show up at all in any sequence of the database DB. In the previous example, 13 out of 64 length-3 candidates do not appear in the database.
An apriory-based sequential pattern mining method adopts a multiple-pass, candidate generation-and-test approach, which creates different challenges, starting with a huge set of candidate sequences that it could be generated in a massive sequence database. This is explained by the generation of a set of candidate sequences, which incorporates all the conceivable stages and redundancy of items in a sequence. The apriory-based approach may produce a truly extensive set of candidate sequences notwithstanding for a moderate seed set.
Example
Another example of database DB includes only two frequent sequences of 1-item, (a) and (b), will generate five candidate sequences of 2-items: (aa), (bb), (ab), (ba) and ((ab)), where ((ab)) is an itemset that represents two events a and b, happening at the same time slot.
Accordingly, if a large sequence database contains 1,000 frequent sequences of 1-item, such as

$(a_1), (a_2), ...(a_{1000})$, an Apriory-based algorithm will generate 1000 * 1000 + (1000 * 999) /2 = 1,499,500 candidate sequences.
However, this support a nontrivial cost of candidate sequence generation, test, and even the support counting is considered as a characteristic belonging to the Apriory-based approach, regardless of what technique is handled to enhance its detailed implementation.
Moreover, the candidate generation-and-test approach go through multiple scans of databases in mining where the length of each candidate sequence increases by one at each database scan. Furthermore, the search space is amazingly extensive. For instance, with n attributes there are n^k possibly frequent sequences of length k. With a large number of articles in the database the issue of I/O minimization gets to be principal. Notwithstanding, most current algorithms are iterative in nature. They acquires the same number of full database examinations as the longest frequent sequence patterns, which is unmistakably an exceptionally costly process.

In case of text analysis, the databases generally include a large amount of sequential patterns and several patterns are extremely long. There is a need for more efficient mining methods. Hence, it is imperative to re-think the sequential pattern mining issue in order to investigate more effective and extensible methods. The next objective is to beat these restrictions. Thereby, there is a need to a new method called Pattern-Growth-based approaches.

2.5.2 Pattern-Growth-Based Approaches

The core of pattern-Growth-based approaches is the separation of the sequential patterns taking into account all the extracted subsequences, and also the arrangement of the sequence database based on the sub patterns.
Pei et al., (2005) define the major idea of the pattern growth as follows:
Instead of projecting sequence databases by taking into account all the possible arrangements of frequent sequences, the complete set of sequential patterns could be partitioned . Given two sequences A and B such that B is a subsequence of A,i.e., $B \sqsubseteq A$. A subsequence A' of sequence A,i.e., $A' \sqsubseteq A$ is called a projection of A. This approach can recursively extend a sequence database into a set of littler databases called projected databases. After that, the projection is based only on local frequent fragments from every projected database known as frequent prefixes because any frequent subsequences can always be found by growing a frequent prefix.
The divide-and conquers pattern-growth principle is adopted by a projection-based sequential pattern mining method, called PrefixSpan where it is considered as one of the efficient and scalable straightforward pattern growth method. PrefixSpan can deal efficiently with large sequence database and also can generate projects databases by growing frequent prefixes.
Key features of pattern growth-based algorithm are:

Search space partitioning: It allows dividing of the generated search space of large candidate sequences for effective memory management. There are various ways to divide the search space. When the pursuit space is divided, littler partitions can be mined in parallel. There are different advanced techniques for search space partitioning such as projected databases techniques.

Tree projection: Tree projection for the most part accompanies pattern-growth algorithms. Therefore, algorithms implement a physical tree data structure representation of the search space, which is then traversed breadth-first or depth-first in search of frequent sequences, and pruning is based on the apriori property.

Depth-first traversal: That depth-first search of the search space has a major effect in performance, thus it helps in the early pruning of candidate sequences. The principle purpose behind this performance is the way that depth-first traversal uses far less memory, more coordinated search space, furthermore less candidate sequence generation than breadth-first or post-order which are used by some early algorithms.

Candidate sequence pruning: Pattern-growth algorithms attempt to use a data structure that permits them to prune candidate sequences right on time in the mining process. This results a smaller search space and maintains a more coordinated and smaller search procedure.

2.5.2.1 Mining Sequential Patterns by Prefix Projection: PrefixSpan Algorithm

PrefixSpan algorithm proposed by Pei et al., (2005)aims to avoid checking every possible combination of a potential candidate sequence by fixing the order of items within each itemset. Because, generally items in any itemset of a sequence are recorded in any order.
PrefixSpan can fix the order of item projection in the generation of a projected database using alphabetic order. For example, the sequence with Sequence-id 2, in the table (2.2) is listed in an alphabetic order <(ad)e(bc)(ae) > rather than <(da)e(cb)(ea) >. With this convention of ordered items, the structure of a sequence is unique.
Intuitively, PrefixSpan examines the projected databases following the order of of the prefix of a sequence and then projects only the postfix of a sequence. By checking the database with a respect to this methodical way, all the possible subsequences and their associated projected database will be extracted.
Definition (Prefix) Suppose all the items are listed alphabetically within an itemset. Given a sequence S = $< e_1, e_2, ..., e_n >$ where each e_i represents a frequent itemset in sequence S, a sequence $S_2 = < e'_1, e'_2, ..., e'_k >$ where k < n, then S_2 is called a prefix of S if and only if, $e'_i = e_i$ for i < m, and ($e'_m \subseteq e_m$) and finally, if all the items in $e_m - e'_m$ are alphabetically ordered after the items within e'_m. For example, < a >, < aa >, < a(ab) >, < a(abc) > are considered as a prefixes of the sequence-id = 10 in (fig1.1). However, < ab > is not considered as a prefix for this sequence.
Definition (Prefix) Given a sequence S = $< e_1 e_2 ... e_n >$ where each e_i represents a frequent itemset in sequence S, $S_2 = < e"_k, e"_{k+1}, ..., e"_n >$ where $k < n$ is the prefix of S.
Sequence S' is a postfix of S regarding the prefix S_2 defined as S' = S / S_2, where $e"_k = (e_k - e'_k)$.
For example, considering the sequence $< a(abc)(acd(cf) >$, we can present the projected database (Table 2.3) as follow:

$Prefix$	Projected (Postfix) database
$< a >$	$< (abc)(acd)(cf) >$
$< aa >$	$< (-bc)(acd)(cf) >$
$< a(ab) >$	$< (-c)(acd)(cf) >$

Table 2.3: Projected databases of a sequence

The S-projected database is the selection of postfixes of the sequence S with regard to the prefix S_2.

2.5.2.2 PrefixSpan Algorithm

The prefix algorithm(see algorithm 2) is based on the prefix and postfix concepts.
PrefixSpan intent to find the complete set of length-1 sequential patterns $< x_1, ..., x_n$ in a sequence database TBS. Then, the complete set of sequential patterns in S can be divided into n disjoint subsets where each subset (1≤ n) represents the set of sequential patterns with prefix x_i.
Given S with a length-m sequential pattern and $< S_{21}, ..., S_{2k} >$ is the complete set of $(m + 1)-$length sequential patterns with regards to prefix S.
The set of all sequential patterns with prefix S, except S itself, can be divided into m disjoint subsets. The collection of j subsets, where (1≤ j ≤ m), is the set of sequential patterns prefixed with S_{2j}.

Example

For example, with the same sequence database DB in table **??** and min-sup_{min}=2, sequential patterns in S can be discovered by a prefix-projection method as follows:
Firstly, PrefixSpan scans DB only one time to discover the complete collection of the frequent items in sequences. Each of these frequent items is a length-1 sequential pattern. They are $< a >$=4; $< b >$=4;$< c >$=3;$< d >$=3; $< f >$=3. After that, all the sequential patterns can be divided into six subsets according to the six prefixes $< a >$; $< b >$;$< c >$;$< d >$; $< f >$. Finally, PrefixSpan finds subsets of sequential patterns. These subsets can be mined by constructing the corresponding set of projected databases and mining each recursively.
Prefix Span unlike the apriori approach, instead of generating candidate sequence, this algorithm only grows longer sequential patterns from the shorter frequent ones, which reduces time consuming and the cost of finding frequent sequential mining.

Algorithm The following algorithm describe the steps of the prefix span. It takes as an input a sequence of database and the minimum support threshold $supp_{min}$ and it returns as output the complete set of sequential patterns.

Algorithm 2 Prefix Span algorithm

Input *A sequence database S, and the minimum support threshold* $supp_{min}$.
Output *The complete set of sequential patterns.*
Method: *PrefixSpan(<>, 0, S)*
Subroutine: *PrefixSpan(* α*, l,* $S|_{\alpha}$*)*
Parameters: α*:a sequential pattern; l: the length of* α*;* $S|_{\alpha}$ *: the* α*-projected database, if* α *!= <>; otherwise, the sequence database S*
Method:
Scan $S|_{\alpha}$ *once, find the set of frequent items i where*
(1) i can be combined with the last element of α *to form a sequential pattern;or*
(2) $< i >$ *can be appended to* α *to form a sequential pattern*
for each
frequent item i, append it to α *to form a sequential pattern* α*', and output* α*';*
for each
α*', construct* α*'-projected database* $S|'_{\alpha}$*, and call PrefixSpan(*α'*,l+1,*$S|'_{\alpha}$*);*
end for each
end for each
END =0

2.5.3 Sequential Pattern Mining Under Multiple Constraints

The previous sections present that regularities easily flow from the frequent patterns. But the minimum frequency constraint is not adapted for all applications. Moreover, the frequency is not the only significant criterion to build interesting user patterns. Other tasks require different constraints to satisfy the user's interest. The Closed Sequential Pattern Extraction under Constraints (see algorithm 3) is proposed by Béchet et al. (2012) is based on a hybrid approach combining different algorithms such as prefixSpan. This algorithm extracts only sequential patterns that satisfy the combination of different syntax and symbolic constraints like the minimal threshold, the length of patterns etc. Notwithstanding to the large scope of these constraints, this method runs on sequences made of itemsets and not only simple items. In the Natural language processing filed context, it implies that a word can be represented by various information such as the categorical grammar of the word, its lemma and even the word itself. This enables to provide more valuable patterns since there is a different level of abstractio.

This method represents the core of our work, for this reason it will be more detailed in the next chapter.

Algorithm 3 Clospec algorithm

Input *SDB: Sequential database C: $\sum$ of constraints.*
Output *The set of sequential patterns under constraints.*
Build all frequent itemsets of 1-length-Patterns verifying C
for *each pattern P of 1-length-Patterns*
Build the projected database of P where infrequent items have been removed
Check constraints C
end for each
end for each
END =0

2.6 Conclusion

In this Chapter, we have introduced the basic notions of pattern mining. We have, also, discussed several works in this field where we have clarified the strengths and the weaknesses of the different methods. In the next Chapter, we will present the proposed approach while detailing different steps.

Part II
Contributions

Part II represents the contribution of this master thesis. This part focuses on presenting our method then it focuses on the experimental protocol and the analysis of the extracted patterns

Chapter 3

Pattern Mining from Textual Data

3.1 Introduction

After discovering the different approaches that contributed to the resolution of this research problem about the extraction of textual features. This chapter is devoted to the description of the research problem (section 3.2), while presenting the method used during this work in order to extract emerging patterns that may correspond to linguistic features (section 3.3), which represent the main objective of our work.

3.2 Research Problem

The main objective of this research project is to provide an additional information to the analysis of linguistic registers by exploiting different textual data. This perspective is based on the extraction of linguistic patterns from texts that aim to characterize the different French register. From a search perspective, we show the interest of developing an approach that extracts both sequential patterns of items (simple elements) or itemsets (set of elements) from textual by passing through the extraction of these patterns from artificial texts. This approach is characterized by the use of different constraints besides the minimal frequency constraint, in order to extract specific patterns of each text, it may represent the characteristic of textual data especially language register in French.

To clarify the research problem, consider the example of sentences:

Example : Given these different sentences in French with English translation between parenthesis:
(1) : *Tu as acheté une belle voiture.*

(You bought a nice car.)
(2) : *Vous venez d'acquérir une somptueuse automobile.*
(You just bought a sumptuous automobile.)
(3) : *T'as acheté une super bagnole.*
(You bought a super carriage.)

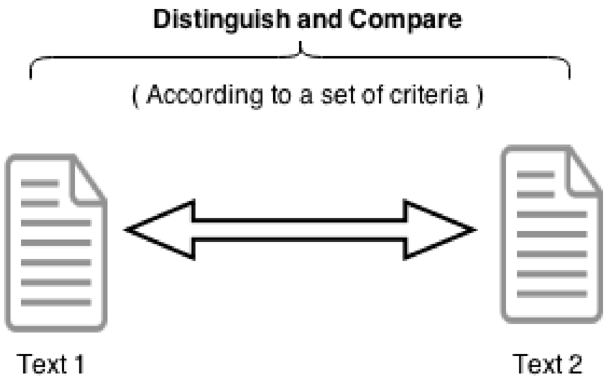

Figure 3.1: General view of the research problem.

Each of these sentences has the same meaning. It is written in different language registers where the first represents a neutral register, the second represents a formal register and the last it is a familiar register. Our goal is to automatically extract the frequent patterns that will represent the different levels of formality. For instance, the sentence (3) can be identified by the recognition of the pattern $< t' >=< tu, contracted >$ that represent only the familiar register. To achieve that, it is essential firstly to detect the sentences that include a specific stylometric feature because it is on those phrases that the approach leans to extract patterns. The identification step constitutes the part of the recognition of linguistic patterns. From there, the method of extraction of sequential patterns can be applied in order to identify text characteristics.

The following section will detail the methodology used in our ongoing work to address the issue of characterizing a text data.

3.3 Methodology

Our approach is split into three parts, as illustrated in the figure (Figure 3.2), namely, Text pretreatment, the extraction of frequent features and the extraction of emerging features. The identification of such features is considered as a challenge in the natural language processing, as their structure is complex.
The sequences are constructed from a text corpus. They are considered as the sentences of the text

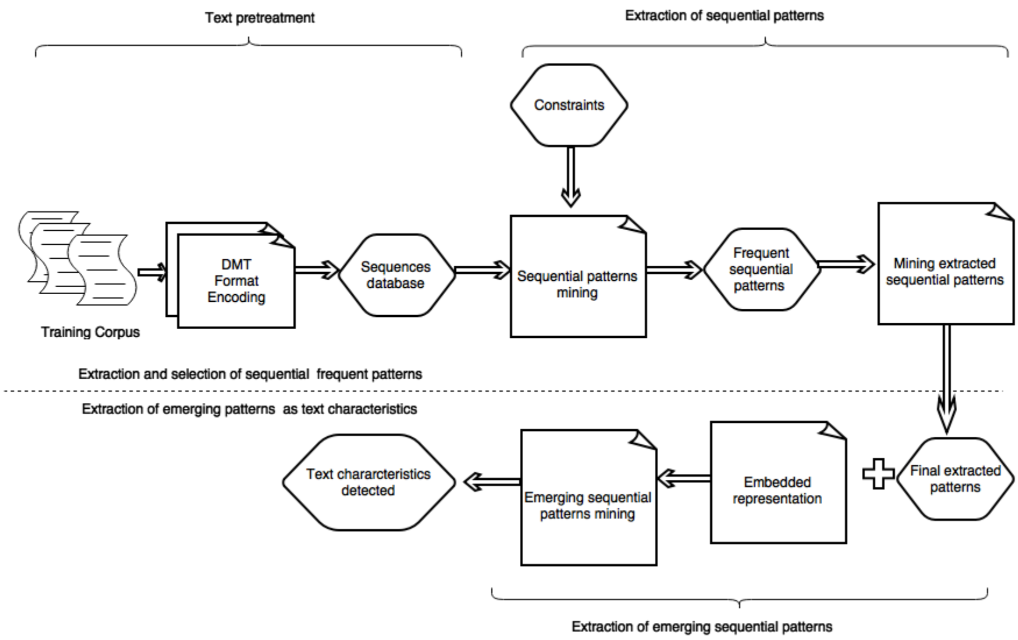

Figure 3.2: General view of the pattern extraction method.

corpus, that may contain the characteristic of a specific language register.
Firstly, each word is replaced with an item set containing the word itself and its format.
Once the sequences base is constructed, we will extract frequent sequential patterns (see subsection 3.3.2) based on multiple constraints that will be detailed in the next section.
Finally, emerging patterns of each corpus are extracted from the sets of sequential patterns presented earlier in a specific representation of classes, in order to discover powerful contrast between different language registers in order to extract automatically specific linguistic characteristics of each language register.

3.3.1 Text Pretreatment

3.3.1.1 Text Corpora

A text corpus is defined as a systematic collection of large naturally occurred texts that follow specific linguistic rule. It represents a core of written or spoken material that may contain texts in a single language (monolingual corpus) or text data in multiple languages (multilingual corpus). Moreover, corpora are considered as the main knowledge base of corpus linguistic fields where corpora analysis defines the subject of different fields like computational linguistic etc. In fact, corpora are used as a type of foreign language such as the contextualised grammatical knowledge, acquired by non-native language user introduction to accurate texts in corpora allows learners to

handle the way of sentence development in the objective language, empowering successful writing. Additionally, the corpus analysis provides in return to grammarians and lexicographers lexical information, morpho-syntactic information and semantic information that bring better descriptions of a language.
Analyzing corpora provides answers to:

- What are the most frequent words and sentences?
- What tenses do people use most frequently?
- What specific patterns are related to lexical or syntactical features ?
- How does these patterns varies regarding different varieties and registers ?

3.3.1.2 Training Text Corpus

We must go through the stage of the generation of artificial text corpus to identify and provide all the sentences that will represent the basis for the extraction of linguistic characteristics. An artificial corpus contains texts from a specific sort. It aims generally to be representative. Artificial corpora can be large or small and it is usually generated in order to answer very specific questions. This corpus could be a boon to plan recognition research, providing a platform to train and test extracted features, as well as allow different features to be compared. In the absence of available textual data, it is necessary to constitute a learning text corpora. Then, three corpora of different sizes are formed. Each corpus consists of more than 10000 expressions (i.e.sequences) that represent at least 53 352 words. The training corpora should be conformed to the rules of a formal grammar. Following figures represent respectively the different automata's representing the basic grammatical rules of each text corpus (L, L1, L2) describing the distinct level of formality.

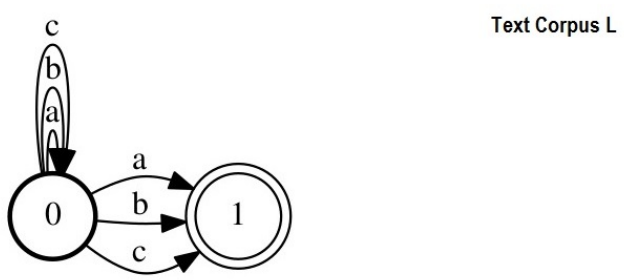

Figure 3.3: Representation of text corpus L.

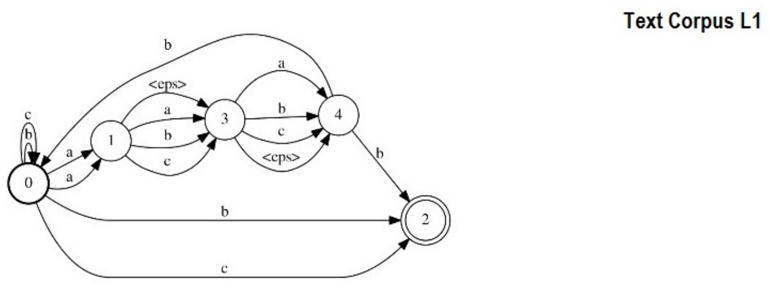

Figure 3.4: Representation of text corpus L1.

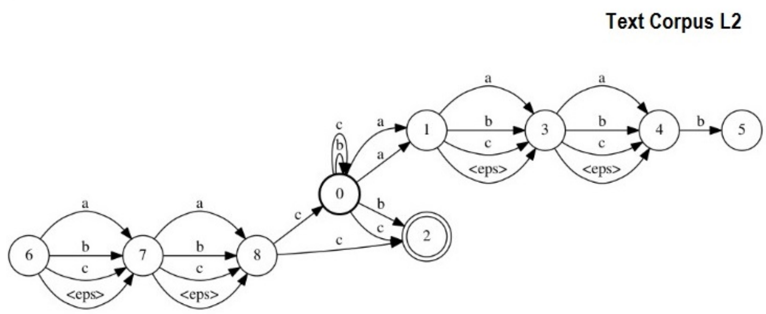

Figure 3.5: Representation of text corpus L2.

L (see Figure 3.3) represents the general artificial language which corresponds to a neutral register. L1 (see Figure 3.4) and L2 represent (see Figure 3.5), respectively the formal and the familiar register where each one of them has some specificity that we attempt to extract them in order to identify each corpus. For instance, each sentence of the three corpora is composed of artificial words (a,b,c), where L is identified by the frequency of $<(a),(b)>$ pattern and L1 is recognized by the pattern $<(a),(c)>$ and finally we can identify L2 by the presence of the pattern $<(b),(c),(a)>$.

3.3.1.3 Data Preparation

The mining of the artificial text corpus is considered as a hard process that makes the extraction of linguistic patterns a really difficult task because of text formatting and size. Henceforth, this approach must go through the stage of text pretreatment in order to identify and provide all the sentences that will be the basis for the extraction of requested patterns. Accordingly, a new text format must be generated based on the text corpus. Then, the sequence database is constructed according to a specific format, called Dmt format. The Dmt format present the extracted items and itemsets over a single sequence or several sequences. The methodology followed to transform text corpus can be described by the figure below (Figure 3.6). To realize this process, the established process goes through a processing chain including two main steps: the extraction and the transformation. The Dmt algorithm aims to extract candidate patterns over a single sequence or several

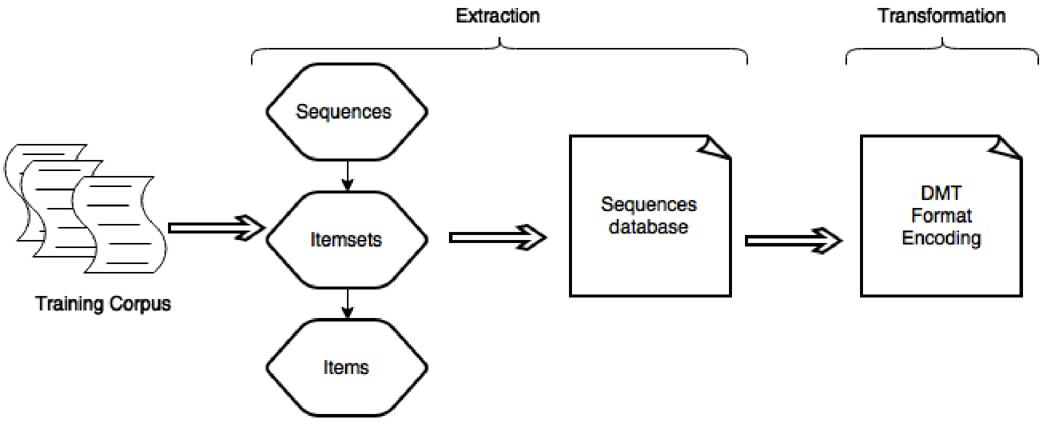

Figure 3.6: Data preparation process.

sequences of events as presented in figure 3.7. Actually, this algorithm comes with elementary sequential data mining features, which tend to extract each element of a specific text corpus in order to transform it into an easily readable format that saves memory by indexing and encoding all items in order to have the possibility to be executed by the sequential pattern mining algorithm. In Dmt format, each sequence: S=$\{e_j\}$ is represented by a series of itemset e_j and each e_j contains i_k where every i_k corresponds to an item. The sequences database (Table 3.1) is constructed based on the training text corpus where each sequence serves as the sentences of the corpus that contain at least one characteristic of a specific language register. In this context, this step refers to the way that the algorithm analyzes a sentence or phrase in terms of grammatical constituents and identify its component parts or items and also transforms the items into itemsets in order to facilitate the extraction and the selection of characteristics.

ID	Sequences
1	$< (b,maj)(b,maj) >$
2	$< (a,min)(b,min)(b,maj)(c,min)(a,maj)(a,min)(b,min)(b,maj)(c,min) >$
3	$< (a,maj)(b,min)(b,min)(c,min)(b,maj)(b,maj)(c,maj) >$
4	$< (a,maj)(a,min)(b,maj)(b,min)(c,min)(c,maj)(c,max)(c,min) >$

Table 3.1: Example of sequence database of training corpora

The resulted patterns by the algorithm are presented in Dmt format as a comma separated list of extracted patterns that corresponds a list of words. Based on the text corpus, every sequence S corresponds to a sentence and each itemset may represent the word and its format.
Hence, the Dmt algorithm saves the results in a text form. Each specific line represents the pattern number; the number of occurrences of the pattern; and also the information about the location of the pattern as a sequence number. For example, considering the following text corpus sentence:

$$\begin{array}{c} \text{SeqId k} \\ e_1 \;\; e_1 \;\; e_1 \;\; \ldots\ldots \;\; e_N \;\; e_N \\ i_1 \;\; i_2 \;\; i_3 \ldots\ldots\ldots\; i_1 \;\; i_2 \end{array}$$

Figure 3.7: Example of Dmt data format

a a c A a c a B C where it is constituted with different items set (a,min) (c,min) (a,maj) (b,maj) (c,maj)
The Dmt sequence corresponding to the previous sentence will look like:

Seq ID 1

item set	2	2	3	3	4	4	5	5	6	6	7	7	8	8	9	9	10	10
item	5	3	5	3	7	3	5	4	5	3	7	3	5	3	6	4	7	4

This Dmt format describes one sentence with 9 words where seqID 1 corresponds to the sequence number; and items correspond to the pattern number and additionally to the sequence number, the itemsets provide an additional information about the location of the pattern.

After that, these results have to be transformed to a text format in order to be interpretable by an expert. Patterns are first rewritten by replacing event numbers by words in order to locate the pattern in the text corpus. For example the pattern (5,3) is rewritten into :(a,contracted). At the end of this step, the Dmt produces adequate results that may help to understand sequences and all their properties. Accordingly, the output is then saved as a text in the right format, including all related details making it easy to use this text for post processing or for analytical purposes, which can be applied to data setted to extract the data requested.

3.3.2 Extraction of Frequent Patterns

The core of the methodology is based on the concept of frequency: the more an ordered list of words appears in the text data, the greater its chance to appear in the results increase. It would be more beneficial in our case through lemmatization. Lemmatization allows to put words in their canonical forms (the word in the right format that respect French rules) in order to make the word more general. In this case, the word will appear more often in the corpus, which induces an increase in the probability they appear in the results.

3.3.2.1 Extraction of Sequential Patterns Under Multiple Constraints

Once the corpus entities identifie and (items, item set, sequence) and the transformation of the corpus completed, the process advances to the second stage. As explained in the previous section, the method is based on the extraction of sequential patterns under multiple constraints. The expected result of this method is to extract at the end the frequent sequential patterns related to stylometric features that will contribute to their extraction in texts.

The following subsection presents the process of the extraction of frequent patterns and it explains the necessity to express a variety of constraints that are able to bring out the quintessence of patterns. **Algorithm Under Constraints** The use of the Closed Sequential Pattern Extraction under Constraints (CloSpeq) algorithm aims to reduce the search space of patterns thanks to the combination of various constraints. This approach allows us to model both linguistic knowledge and to filter the most relevant patterns depending on the research problem.
The goal of this approach is to obtain sequential patterns that reflect certain linguistic regularities and the discovery of stylometric features in the text corpora.

The expression of the constraint summarizes the expectations of the user. Although it is almost impossible to draw up a complete list of constraints used in the literature (Agrawal and Srikant, 1994; Ng et al., 1998; Kiefer et al., 2003; Bonchi and Lucchese, 2005), there are several major categories. In fact, constraints allow to incorporate domain knowledge. Therefore, they enable to discover and to select only the patterns satisfying these constraints to reduce the redundancy of sequential patterns. These are detailed below:

The Frequency constraint: requires the condition that each extracted pattern must be frequent. Consider the following example: the pattern S $=< (a)(e) >$, presented in the table (Table 3.2) where sup_{min} equals to 2. For this example, $sup_S = 1$, from thus S is not a frequent pattern and in this case it will not be extracted and it is the same for its superior patterns. This is because of the following property: If $S_1 \leq S_2$ then $sup_{S_1} \geq sup_{S_2}$. The choice of the minimum support threshold is a recurring problem in data mining. If the support threshold is too high, the risk is to extract only generalities that will not allow discovering anything to the user.
However, if the support threshold is too low, the set of extracting frequent patterns can be extremely large, making it impossible to use. The chosen option is to consider a low support threshold in order to preserve the maximum information, including infrequent patterns in texts, but it may be relevant, and reduce all frequent sequential patterns by introducing additional constraints and also the use of recursive search process.

ID	Sequences
1	$<(de)(abc)(df)>$
2	$<(ab)(de)(cf)>$
3	$<(e)(ab)(df)>$

Table 3.2: Example of sequence database from training corpora

The Gap constraint: aims to check the contiguity between itemsets. In other words, it puts a condition regarding the number of items that can exist between two neighboring itemsets of a candidate pattern by considering all sets of sequences.
The gap must be included in an interval $[M, N]$ from which the annotation $S_{[M,N]}$. For example, $S_{[0,1]} =< (e)(f) >$ appears three times in the sequence database table, in S_1, S_2 and S_3 because the amount of items allowed between (e) and (f) is either 0 or 1. For against, $S_{[1,2]} = < (a)(f) >$ appears only once, in S_1 because in this case, the Gap is included between 1 and 2.

The length constraint: with this constraint, the user has the possibility to limit the number of itemsets constituting the extracted pattern. For example, if the length of the pattern is between 3 and 6 then S' = $< (a)(bce) >$ will not be extracted because it only contains 2 itemsets (length of pattern= 2).

The arising challenge of this approach is to consider all constraints. Indeed, the introduction of Gap may invalidate the anti-monotony of support. Therefore, the authors propose an efficient algorithm to respect all these constraints.

The algorithm calculates at first the frequency patterns of a single item that will contribute to the construction of a projected database. For this purpose, the algorithm extract from the sequence database all suffixes of these patterns that are respectively considered as prefixes in the sequences. This projected database is built with respect to the frequency constraint and will disclose a new sequence database from which the algorithm will calculate this time frequent patterns containing two items and build on this, another projected database from these new frequent patterns. And so on, until then total reduction in the size of the projected database.

Combinations of constraints So far the proposed constraints are defined by a single selection criterion. The atomic constant term is then privileged to designate it. In the following, the term refers to a combination of forced atomic constraints. The combination of these constraints is considered itself as an important constraint because it may even enrich the expressiveness of the extracted patterns. If an atomic constraint is insufficient to express the nature of research patterns, then the user can complete it with one or more other criteria to refine its expectations. The combination of atomic constants allows to combine their respective semantics.
In particular, this allows the extraction of patterns adequate to the semantics of each atomic constraint. For example, the combination between frequency constraint and the length constraint: freq(X) ≥ min-supp $\wedge Nb-itemset \leq$ Max-itemset.In addition to targeting interesting information, this combination of constraints reduces the number of extracted patterns and thus facilitates their analysis.

Our work is income from the extraction patterns originally proposed by Agrawal et al., (1994) has been widely illustrated in the second chapter. The pattern mining represents a research field for which there is connections with many other areas. Moreover, the extraction patterns under constraints can be seen as a special class of constraint satisfaction problems in the linguistic field. The patterns are generated from left to right, until the last extension. This process runs recursively to verify if each generated pattern respects the multiple constraints.

3.3.3 Extraction of Emerging Patterns

After extracting the different frequent patterns from different corpora, we attempt to present these set of patterns in a class representation in order to facilitate the extraction of emerging patterns.

3.3.3.1 Representation of Patterns

A major problem in the process of knowledge discovery data is the significant number of producing patterns, making them difficult to be used. Thus, the most significant patterns are often embedded in the middle of trivial or redundant information. This problem is often encountered by local patterns extractors. The luck of expressiveness represents a limit for exploring more relevant patterns. Thus, we need to focus on reducing the number of extracted patterns to get more pertinent patterns.

In practice, the amount of frequent sub sequences along with their supports can be very large. There is an interesting way to decrease this number and to eliminate the redundancy without loss of information, which is to embed all the extracted patterns from different text corpora into only one text corpus with the specification of the source of each pattern through the use of classes. Accordingly, this representation intents to construct powerful classifiers in order to help setting up diagnosis. Ideally, this representation (Figure 3.9) is much helpful than the original set of frequent patterns and it can be extracted more efficiently, while permitting a speedy mining of the whole set of frequent patterns without scanning different text corpus at the same time that is considered as a costly scan of the text corpora and new support counting.
Another interesting achievement is that this representations are not only useful for frequent sequential mining in difficult cases, but also to derive more meaningful patterns. In other terms, instead of a extract patterns from different text corpora, which can be impossible due to the size of the collection of frequent itemsets, it is possible to use directly this representation to facilitate the mining of emerging patterns from one data set that contain all the useful information. In other words, instead of applying the pattern mining algorithm on each corpus, then compare the different results of corpora. Accordingly, it allows to detect differences and similarities between patterns of different texts corpus, which it characterizes the classes in a quantitative and qualitative manner.

Let DB be an example of our artificial training corpus presented in table 3.3.

Each line of this table represents a set of features or items (a,b,c,d,e) and both C1, C2 describe class values. DB is divided into two data sets D1 and D2 which represent two different text corpus. The transactions having item C1 (resp. C2) correspond to D1 (resp. D2). A transaction t_i contains

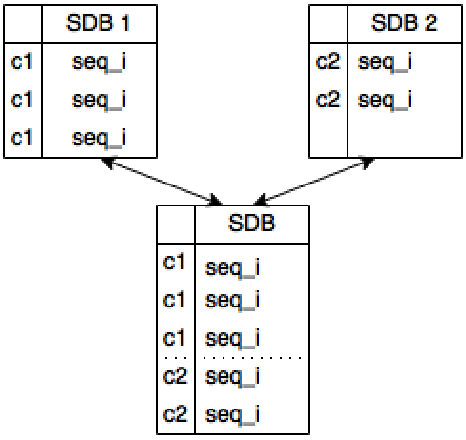

Figure 3.8: Sequential pattern representation example.

Sequence − id	Items	class
1	a b c d	C1
2	a b c	C1
3	a d e	C1
4	a b c	C2
5	b c d e	C2
6	b e	C2

Table 3.3: Example of patterns representation

a pattern X if and only if X ⊆ t. Since, the main idea of emerging patterns is related to the notion of frequency. The frequency of a pattern X in a data set D noted F(X, D) is the number of transactions of D that contain X. The main advantage of this representation is to facilitate the extraction of emerging patterns which may correspond to language register characteristics. Let get the growth rate of any pattern X, it is necessary to compute F(X, D) and F(X, Di). These frequencies can be obtained from the this representation of frequent patterns, which helps to determine if a pattern in C1 (resp. C2) is more frequent in C2 (resp. C1), which defining the emerging pattern.

3.3.3.2 Emerging Sequential Pattern Mining

As part of the pattern analysis, the used representation in the previous subsection 3.3.3.1 contains frequent sequential patterns from the different text corpora that correspond to a different language registers. These are the extracted pattern during the previous step of our approach (see subsection 3.3.2.2).

Emerging patterns are calculated using the equation where L_1 conforms to one class which

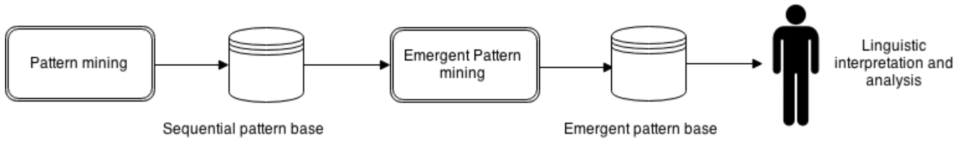

Figure 3.9: Emerging patterns process.

corresponds specific language register and L_2 to another class and another language register. Emerging patterns use the classes to discover the difference between an amount of classes.

$$\text{GrowthRate(GR)=}\begin{cases} \infty, & if\, supp_{L1} = 0 \\ supp_{L2}/supp_{L1}, & else \end{cases}$$

The emergence of a pattern is quantified by measuring its growth rate (GR) between two classes, otherwise it may tend to infinity if patterns do not exist in one of the classes.
Link between Emerging Sequential Patterns and Linguistic characteristics Extracting sequential patterns allow the automatic extraction of sub-sequences from a set of sequences by combining their different items while respecting the different constraints that can be set according to the desired results. In our case we tend to automatically extract linguistic patterns that lead the discovery of the linguistic characteristics of different language registers. Patterns are a combination of grammatical categories or words formats and words. For example, the pattern $< tu, contracted >$ $< Verb > < pas >$ may represent a relevant pattern to identify the familiar register regarding the contraction of certain terms like negation marker *(ne + Verb + pas ⇔ Verb + pas) and apocope of the subject tu (tu ⇔ t').* Therefore, the extraction of language register markers is based on this context. In other words, the sequences will be described by itemsets comprising their items.

3.4 Conclusion

In this Chapter, we have elucidated how emerging patterns could be used to select the language register features. we have presented an extraction method at extracting the right linguistic markers with respect to a set of constraints while reducing search space using classes of frequent patterns.

Despite the noticed advantages of our proposed methods, they have some limitations since they may generate noise and thereby this may converge to undesirable results. This will be dealt with in the next Chapter.

Chapter 4

Experimental Study

4.1 Introduction

In this chapter, we will present the results of our experimental evaluation. Various tests were carried out to understand the interest of parameters for the sequential data mining, including constraints. We will describe first the used corpus and the setting of different parameters used while extracting emerging patterns. Moreover, we will present an analysis of the extracted patterns in two parts: a quantitative perspective and a linguistic perspective.

4.2 Pattern Extraction

Sequences of the SDB are the representation of the sentence of the different training corpora, with different sizes: each corpus may have 100, 1000, 10000 sentences. In the sentence each word is replaced by either an item or an itemsets. However, the amount of the extracted patterns appears to be important, even with the use of certain constraints that tend to reduce the search space. For example, the constraint of minimal support is very relevant to prune the patterns. Following the property «any higher pattern contain a non-frequent pattern is not frequent»will reduce the number of extracted patterns while reducing at the same time the computational complexity.

The experiment protocol is the way to test extracted patterns. It will be the way to evaluate the impact of each constraint on the extraction of emerging patterns. During experiments, we use usual constraints, for example, the minimal frequency constraint and other useful constraints that convey some additional linguistic information like the gap and the minimal length constraints, in order to discover accurate emerging sequential patterns while reducing the search space. These multiple constraints enable us to express a large scope of knowledge to focus on interesting patterns The main objective of our approach is to extract emerging sequential patterns, that express linguistic regularities such as characteristics of language register. The different selected methods provide a possibility to combine at the same time these constraints coming from various origins.

4.2.1 Parameters for Pattern Mining

We consider the sequence, including items or itemsets representing word forms. In order to execute the data mining techniques on the different corps, we perform, in the first place, the dmt4 transformation task; this allows us to define several constraints on extracted patterns, for example, their length and their frequency and also it allows the generation of itemsets.

we set the different parameters as follow:

The minimal frequency: We set the value of minimal frequency empirically to be a compromise between extract interesting patterns with low support and thus gives a lower value to minimal frequency. Two values of minimal frequency have been experimented 2 % and 5%.

The minimal length: The goal of this constraint is to remove sequential patterns that are too small with respect to the number of items or itemsets (number of words) to provide relevant linguistic patterns. We set the minimal length value on one hand to 2 words and on the other hand, we tested the case without this constraint, which means without minimal size value.

The maximal scope: This constraint relates to the maximal value of the linguistic scope of a pattern. We tested this constraint with a maximal scope value set of 15 words. It implies that the maximal number of items or itemsets between the first item and the last item of the pattern is 15 corresponding to 15 words in the sentence.

The gap: As regards this constraint, we considered different values, during the experimental protocols, between 0 and 5. We tested the following different combinations [0,0], [0,1], [0,2], [0,3] and [0,5] and without this constraint.

The length of corpus: The experimental study is made with different training corpora where each one of them has different size 100, 1000 or 10000 sentences. Each corpus will be compared to another one from the same size according to a growth rate, in order to extract the emerging patterns that will be represented in linguistic point of view in section 4.4.

The growth rate: Since the different training corpus has different sizes, we conducted experiments with a growth rate threshold set to 2 in order to extract the emerging pattern. This means that only the patterns appearing in more than two times are kept for specific language register. This threshold is used for items and also for itemsets pattern mining.

4.3 Quantitative Analysis of the Patterns

This section presents experimental results of items patterns and itemsets patterns, while comparing two different corpora, showing the impact of the set of constraints. The amount of extracting patterns is considered important, this quantitative analysis may allow to select the different patterns that will be analyzed, from a linguistic point of view.

4.3.1 Impact of Constraints

Table 4.1 indicates the number of extracted sequential patterns with the variation of minimal frequency ($supp_{min}$). Additionally, the reduction percentage of extracted patterns made by the variation of minimal frequency from 2% to 5% are also given for each type of patterns.

	Minimal	frequency	Reduction (%)
Texts	2%	5%	From 2% to 5%
100	5868	889	84.85
1000	4037	949	76.49
10000	4937	941	80.94

Table 4.1: Number of extracted sequential patterns with the variation of minimal frequency constraint.

Based only on the minimal frequency constraint, the results show a large amount of extracted patterns where $supp_{min}$ equals to 2% compared to the results with $supp_{min}$ equals to 5% with the variation of the size of the corpus. Thus, the selection of the best pattern candidates may consume time. Therefore, this allows to infer that this large set of extracted patterns may be too general. Thus, the extraction of language register's characteristics with this type of patterns may generate noise and thereby this may converge to an undesirable result.

Furthermore, both table 4.3 and table 4.2 present the amount of the extracted sequential patterns with respect to several constraints: different gap values and minimal frequency. However, table 4.3 does not take into account the length constraints unlike the table 4.2.

Item Mining: The extracted patterns without length constraint:

	Items	patterns	with	Gap constraint		
Corpus length	[0,0]	[0,1]	[0,2]	[0,3]	[0,5]	No Gap
100	104 (80.2%)	345 (61.92%)	682 (23.52%)	797 (10.45%)	854 (3.98%)	889 (0%)
1000	176 (80.2%)	345 (61.92%)	758 (23.52%)	893 (10.45%)	945 (3.98%)	949 (0%)
10000	88 (80.2%)	335 (61.92%)	675 (23.52%)	846 (10.45%)	929 (3.98%)	941 (0%)

Table 4.2: Number of item patterns with respect to three constraints: $supp_{min}$, gap and without length constraint.

Item Mining: The extracted pattern with length constraint:

	Items	patterns	with	Gap constraint		
Corpus length	[0,0]	[0,1]	[0,2]	[0,3]	[0,5]	No Gap
100	95 (80.2%)	336 (56.92%)	673 (20.52%)	788 (7.45%)	845 (0.98%)	880 (0%)
1000	167 (77.2%)	336 (61.92%)	749 (20.52%)	884 (7.45%)	934 (0.98%)	940 (0%)
10000	79 (77.2%)	335 (61.92%)	664 (20.52%)	838 (7.45%)	920 (3.98%)	932 (0%)

Table 4.3: Number of item patterns with respect to three constraints: $supp_{min}$, gap and length constraint.

The variation of the minimal frequency appears to have the most important effect on the reduction of the number of patterns where the ratio of reduction of extracted patterns over the variation of $supp_{min}$ from 2% to 5% is between 76% and 85%. The two other constraints have a rather minor impact on the number of patterns where the reduction caused by the length constraint is around 0.3%, comparing with the results showed in table 4.3.

The impact of gap constraint depends on the choice of the set values. It can reduce greatly the amount of percentage by 80% when the value of gap is small, comparing to the results of minimal frequency presented in table 4.1. However, by increasing the value of the gap, the reduction's impact of the number of patterns becomes less interesting, since it decreases more and more. This constraint allows to leave a gap, set by the user, between the items of a candidate pattern. In our approach, it allows to ignore certain words in the sentences when extracting the patterns.

The results of item mining highlight the impact of constraints. In the next analysis, we will fix the constraints during quantitative analysis of itemset patterns.

Itemset Mining: The extracted patterns with length constraint:

For the itemset mining, we will fix the length of the corpus on 10000 sentences while continuing the test by setting minimal frequency constraint to 5% and the length of the pattern between 2 and 15.

	Itemset	patterns	with	Gap constraint		
Corpus length	[0,0]	[0,1]	[0,2]	[0,3]	[0,5]	No Gap
10000	9013	163080	790187	1515809	2225033	2475413

Table 4.4: Number of itemsets patterns with respect to three constraints: $supp_{min}$, gap and length constraint.

Based on this example (Table 4.4), the results show that the search space is reduced while decreasing the gap values. This confirms the same conclusion as the previous example of items mining.

The selection of emerging patterns:

Table 4.5 shows the number of extracted emerging patterns from the training corpora, considering the different constraint, the minimal frequency threshold fixed on 5%, with length value set between 2 and 15 items or itemsets and also with different values for the gap constraint. The ratio of emerging patterns over the previous results of extracted patterns under constraints is also given for each type of patterns.

	patterns	with	Gap constraint			
Patterns	[0,0]	[0,1]	[0,2]	[0,3]	[0,5]	No Gap
items	23 (29.11%)	89 (26.57%)	152 (22.89%)	177 (21.12%)	194 (21.09%)	195 (20.82%)
itemset	764 (91.52%)	10866 (93.34%)	35438 (95.52%)	53708 (96.46%)	68087 (96.94%)	69192 (97.2%)

Table 4.5: Number and ratio of emerging patterns with respect to three constraints: $supp_{min}$, gap and length constraint.

Thus, among the extracted patterns from the training corpus with length 10000 by fixing the gap constraint [0,0], 29.11% of them patterns are emerging patterns. Considering only the emerging patterns, the number of patterns to be analysed decreases tremendously. This allows to focus on the most specific patterns for given texts. In the following analysis, we focus on emerging patterns in order to show the contrast between different texts that allows to identify different language register in French. We also note that by increasing the gap constraint, the ratio of emerging patterns of items and itemset tends to increase: this means that these additional emerging patterns are rather non-specific patterns of language register.

Distribution of emerging patterns according to their length:

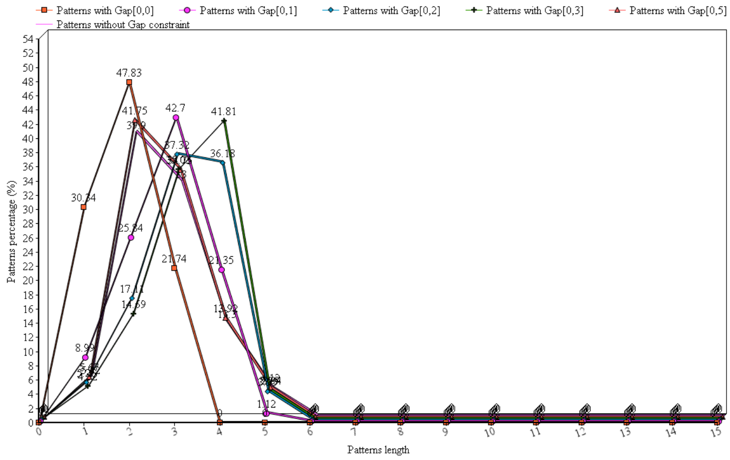

Figure 4.1: Distribution of emerging items according to the length of pattern.

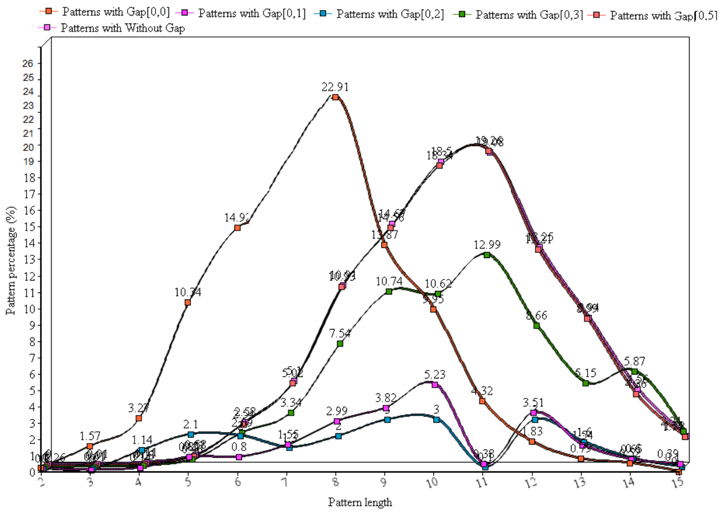

Figure 4.2: Distribution of emerging itemset according to the length of pattern.

We will study the distribution of emerging patterns according to their length. The line graph shown in figure 4.1 and 4.2 gives the relative number of patterns according to their length for both items and itemsets. We see in figure 4.1 that the majority of items of the patterns contain from 1 to 5 items. In addition, the other patterns are considered as rare patterns and less informative.

Otherwise, as presented in figure 4.2, the majority of length of patterns vary between 4 and 11 itemset.

Distribution of emerging patterns according to their growth rate:

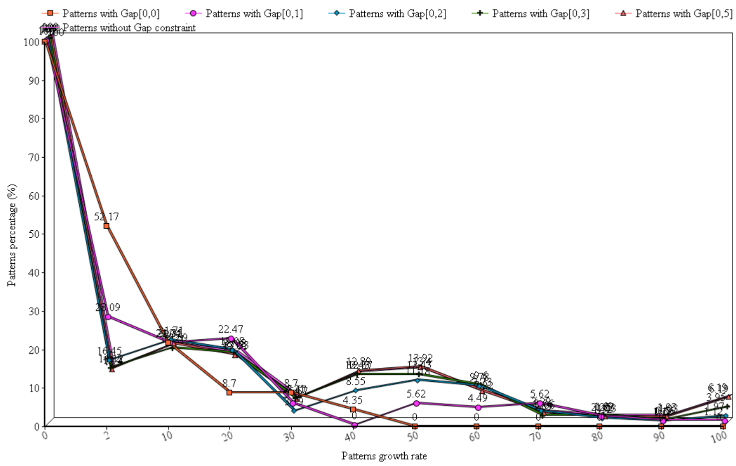

Figure 4.3: Distribution of items according to the growth rate of pattern.

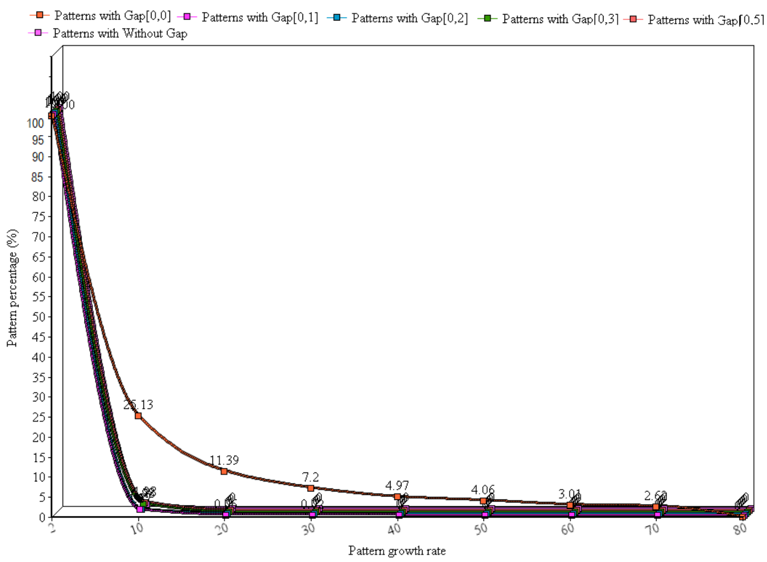

Figure 4.4: Distribution of itemset according to the growth rate of pattern.

Finally, we study the distribution of emerging patterns according to their growth rate. The graph line shown in Figures 4.3 and 4.4 provides the cumulative relative number of all emerging patterns according to their growth rate, for the different corpus sets under multiple constraints where minimal frequency is equal to 5% and the size of patterns vary between 2 and 15.

Based on the example of item mining (Figure 4.3), all the emerging patterns of items have a growth rate greater than 2 and more than 47.84% of emerging patterns of items is grater than 10 even with variation of gap constraints by setting the growth rate threshold = 2 on the threshold of growth. We can observe that emerging patterns start having an infinite growth rate since the cumulative number of emerging patterns remain constant for growth rates greater than 70. This means that this set of emerging patterns appears only in one kind of text (and never in the other). This means that by increasing the growth rate, the linguistic patterns get to be more specific to a kind of text.

In the itemset pattern mining example (Figure 4.4), the majority of emerging patterns has a growth rate between 2 and 10. After this threshold, all the emerging patterns approach to zero. This means that, in this corpus, these sets of patterns exist in both corpora, which indicates that the emerging patterns between 2 and 10 are the set of patterns specific only to one corpus.

Rare Patterns:

Despite our satisfactory results, there still a number of emerging patterns that are considering rare. The table below 4.6 presents the amount of rare patterns.

		patterns	with	Gap constraint		
Patterns	[0,0]	[0,1]	[0,2]	[0,3]	[0,5]	No Gap
items	7 (30.01%)	23 (25.84%)	28 (18.42%)	33 (18,64%)	40 (20.61%)	39 (20)
itemset	164 (21.46%)	1253 (11.53%)	2359 (15.02%)	3653 (14.7%)	4131 (16.48%)	4006 (17.27%)

Table 4.6: Number and ratio of rare emerging patterns with respect to three constraints: $supp_{min}$, gap and length constraint.

Since the extraction of patterns based on the frequency and the other constraints has a rich vocabulary, the ideal would be to increase the probability of occurrence of words may be related to French language register's characteristic in patterns. The proposed idea is to make a generalization of the extracted patterns. In other words, we will add another measure in order to sort these patterns according to their generality by combining the growth rate value with the support frequency (GR $\bigwedge$ Freq) of each emerging pattern.

4.4 Linguistic Analysis of the Emerging Patterns

In this section, we present a linguistic analysis of some extracted emerging patterns. We focus our attention more particularly on finding the correspondence of extracted emerging patterns with the familiar register in French. First of all, we consider single-item patterns. By studying them, we can find some interesting patterns, characteristics of familiar register. Table 4.7 shows examples of such emerging patterns, both in neutral and familiar register in French. In the patterns, the symbol * is used to represent a gap of one or more words. Furthermore, we also illustrate each pattern with examples of underlying sequences in familiar register comparing to a neutral register.

Emerging Patterns in neutral register	Emerging Patterns in familiar register	Neutral Register	Familiar Register
a b c	a c b	*Donne-le-moi*	*Donne-moi-le*
a b c	a c	*Elle n'est pas triste*	*Elle est pas triste*
a b c	b c	*Il fallait faire*	*Fallait faire*
a b c d	a b c b d	*Il fallait faire*	*Fallait faire*
a b c	d b c	*ça ne m'interesse pas*	*Cela m'interesse pas*
(A)(b,*)(*,*)	(a,maj)(b,*)(*,*)	*Tu es petit*	*T'es petit*
(a,min)(*,*)(d,min)	(a,min)(*,*)(d,maj)	*La robe que j'ai mise*	*La robe que j'ai mis*

Table 4.7: Correspondence of extracted emerging patterns with real text.

The extracted patterns allow the observation of schematic lexical, phonetic and morphological phenomena as illustrated in table 4.8 while identifying at each time the corresponding characteristics of language register.

Emerging Patterns in neutral register	Emerging Patterns in familiar register	Correspond Characteristics	Category
a b c	a c b	Anastrophe	Morphological phenomena
a b c	a c	Negation without «ne »	Morphological phenomena
a b c	b c	Suppression of the pronoun «il »	Morphological phenomena
a b c d	a b c b d	Suppression of the pronoun «il »	Morphological phenomena
a b c	< − > d b c	Replacement of words	Lexical phenomena
(A)(b,*)(*,*)	(a,maj)(b,*)(*,*)	Suppression of the pronoun «tu »	Phonetic phenomena
(a,min)(*,*)(d,min)	(a,min)(*,*)(d,maj)	Grammatical conjugaison	Morphological phenomena

Table 4.8: Correspondence of extracted emerging patterns with language register characteristics.

4.5 Conclusion

Throughout this chapter, we focused on the experimental protocol on our data and the appropriate observations were noted. Then, we have conducted these experiments and analyzed results showing the impact of conjunctions of various syntactic and symbolic constraints while evaluating their advantages as the number of emerging patterns. Finally, we show the link between our data and language registers in French. Our method gave satisfactory results in detecting and extracting emerging patterns. However, it had some undesirable results that represent rare patterns.

Conclusion

The extraction of information in the texts represents the core of the work presented in this master thesis. If the extraction of information has generated interest in recent years, it is for the important role that it could have on different fields. Since the object of this study is the language register, this theme is approached from a linguistic angle in natural language processing and data mining. In our case, the work done was based on the extraction of linguistic patterns of different textual documents in order to develop an efficient approach that aims to the extraction of the linguistic characteristics of language register.

This thesis report is rooted precisely in this context to carry out this research. We had present and describe the approach used in our ongoing work. The latter is classified as an unsupervised data mining method incorporating the specificity of the natural language processing field because of its principle, which is based on a search of sequence data, specifically, the extraction of sequential patterns. We have indeed shown the interest in using the extracted patterns from the data mining methods from textual data, especially in information retrieval tasks. Indeed, extracted patterns will represent linguistic patterns that will allow us to extract linguistic characteristics of language registers in French. We have proposed a method to extract emerging patterns, including different types of information and producing ordered patterns in a hierarchy (from the most general to the more specific). On the search plan, two approaches have been designed to provide a solution to the problem by the abundance of producing patterns and the selection of the most interesting one. One is to incorporate the language skills in the mining process under constraints. The second is to use classes to facilitate the search of contrast between different texts.

However, in order to get the desired results, the proposed approach requires multiple steps. The first step attempts to pretreat the training text corpora which represent textual data in order to facilitate the recognition and the identification of sequential patterns. This step is based on the extraction of the different component of the sentence, hence, their transformation into an interpretable text on a specific text format, known as Dmt format.

The next step focuses on the extraction of sequential patterns. Regarding to the large search space of sequential patterns. We proposed to extract these patterns according to a set of combined constraints like minimal frequency to allow the extraction of only frequent patterns, the gap constraint to check the contiguity between words, etc. After mining all different corpora, the extracted patterns were presented into different classes, where each class corresponds to a different text corpus. This step helps the extraction of emerging patterns while comparing the patterns of each class.

Once all steps were completed, we tested the approach on the different text corpora. We have proven the importance and the impact of the set of different constraints on pattern mining process that allows the reduction of the search space with the least possibility of rare patterns. Finally, we have shown the link between the emerging patterns and the linguistic patterns which represent the language register in French.

Our research opportunity lies, of course, in the extension of our ongoing work: Extraction of information and data mining. Indeed, there is several interesting future works that have to be mentioned.

One of the prospects which should not be overlooked is the validation of these linguistic patterns by experts, which probably allow us to validate more patterns and thus extract other characteristics.

As an other extension, we propose to use supervised methods such classification in order to compare the results with those obtained in our ongoing work. Other constraints would be taken into account in order to get more interesting information and then to integrate them into the algorithm with respect to all defined constraints.

Finally, it is possible to develop other innovative applications requiring knowledge of humans, such as opinions, states and competencies by analyzing all the extracted patterns that represent the expressivity used by the authors of the texts.

References

P Achard. Registre discursif et énonciation : induction sociologique à partir des marques de personne. *Le Congrés des Députés du peuple d'URSS*, pages 5–34, 1995.

R Agrawal and R Srikant. Fast algorithms for mining association rules. *Proceeding VLDB '94 Proceedings of the 20th International Conference on Very Large Data Bases*, 1215:487–499, 1994.

J Bailey. Fast algorithms for mining emerging patterns. *Principles of Data Mining*, pages 39–50, 2002.

B Barber and H Hamilton. Extracting Share Frequent Itemsets with Infrequent Subsets. *Data Mining and Knowledge Discovery*, 7:153–185, 2003.

N Béchet, P Cellier, T Charnois, and B Crémilleux. Discovering linguistic patterns using sequence mining. *13th international Conference CICLing)*, 7181 LNCS(PART 1):154–165, 2012.

N Béchet, P Cellier, T Charnois, and B Crémilleux. Sequence Mining under Multiple Constraints. *13th international Conference CICLing)*, 2012.

G Bennett. Using Corpora in the Language Learning. page 4, 2010.

D Biber. A corpus-driven approach to formulaic language in English: Multi-word patterns in speech and writing. *International Journal of Corpus Linguistics*, 14:275–311, 2009.

N Blaylock and J Allen. Generating artificial corpora for plan recognition. *User Modeling*, pages 151–151, 2005.

F Bonchi and C Lucchese. Pushing Tougher Constraints in Frequent Pattern Mining. *Advances in Knowledge Discovery and Data Mining*, pages 114–124, 2005.

A Bykowski and C Rigotti. A condensed representation of frequent patterns for efficient mining. *Information Systems*, 28:949–977, 2003.

D Candel and P Lafon. Approche lexicale des registres en langues de spécialité. *Meta: Journal des traducteurs*, 1994.

M Cembalo and H Holec. Les Langues Aux Adultes: Pour Une Pédagogie De L'Autonomie. *Mélanges Pédagogiques*, pages 1–10, 1973.

C Chand, A Thakkar, and A Ganatra. Sequential Pattern Mining : Survey and Current Research Challenges. *International Journal of Soft Computing and Engineering (IJSCE)*, 2012.

T Charnois. *Vers une hybridation fouille de données et traitement automatique des langues*. 2012.

T Charnois, Plantevit, M, and Rigotti. Fouille de données séquentielles pour l'extraction d'information dans les textes. *Tal*, 2009.

Y Chen and G Lee. An efficient projected database method for mining sequential association rules. *5th International Conference on Digital Information Management, ICDI*, 2010.

G Dong and J Li. Efficient mining of emerging patterns: discovering trends and differences. *Proceedings of the fifth ACM SIGKDD international conference on Knowledge discovery and data mining*, 1999.

G Dong and J Li. Mining border descriptions of emerging patterns from dataset pairs. *Knowledge and Information Systems*, (2):178–202, 2005.

R Duval. Transformations de représentations sémiotiques et démarches de pensée en mathématiques. *Actes du XXXIIe colloque de la COPIRELEM*, 2006.

Y Road. Mining emerging patterns from time series data with time gap constraint. 2011.

U Fayyad, G Piatetsky-Shapiro, and P Smyth. From data mining to knowledge discovery in databases. *AI magazine*, 17:37–53, 1996.

M Gamon and A Grey. Linguistic correlates of style : authorship classification with deep linguistic analysis features. *Proceedings of the 20th International Conference on Computational Linguistics*, 4:611, 2004.

M García-Borroto, J Martínez-Trinidad, and J Carrasco-Ochoa. A New Emerging Pattern Mining Algorithm and Its Application in Supervised Classification. *Lecture Notes in Computer Science*, 6118:150–157, 2010.

A Genkin and D Lewis. Author Identification on the Large Scale. *In Proc. of the Meeting of the Classification Society of North America*, 2005.

C Giannella, J Han, X Yan, and P Yu. Mining Frequent Patterns in Data Streams at Multiple Time Granularities. *Next generation data mining*, pages 191–212, 2003.

N Grover. Comparative Study of Various Sequential Pattern Mining Algorithms.

2014.

J Han. From sequential pattern mining to structural pattern mining: a pattern growth approach . *Journal of computer sciences and technologies*, 2004.

B Habert and P Zweigenbaum. Classer les mots: sémantique à gros grain et méthodologie harrissienne. *Revue de Sémantique et Pragmatique*, pages 25–45, 2003.

J Han, G Dong, and Y Yin. Efficient mining of partial periodic patterns in time series database. *Proceedings 15th International Conference on Data Engineering*, 1999.

F Heylighen and J Dewaele. Formality of Language : definition , measurement and behavioral determinants. *Interner Bericht*, 1999.

J Houvardas and E Stamatatos. N-gram feature selection for authorship identification. *Artificial Intelligence Methodology Systems and Applications*, 2006.

F Iqbal, R Hadjidj, B Fung, and M Debbabi. A novel approach of mining write-prints for authorship attribution in e-mail forensics. *Digital Investigation*, 5(SUPPL.), 2008.

M Jaillet,S , Laurent, A, Teisseire. Sequential Patterns for Text Categorization. *LIRMM-CNRS-Universite*, 2004.

M Khiari and A Lallouet. Extraction de Motifs sous Contraintes Quantifiées. 2013.

S Kim, H Kim, T Weninger, and J Han. Authorship classification - A syntatic tree mining approach. *SIGKDD UP Workshop*, 2010.

S Kim, H Kim, T Weninger, and J Han. Authorship Classification : A Syntactic Tree Mining Approach Categories and Subject Descriptors.

E Knox and R Ng. Algorithms for Mining Datasets Outliers in Large Datasets. *24th International Conference on Very Large Data Bases*, 1998.

M Koppel and J Schler. Exploiting Stylistic Idiosyncrasies for Authorship Attribution. *IJCAI'03 Workshop on Computational Approaches to Style Analysis and Synthesis*, 2003.

T Lenaour. Optimization of manifold learning techniques for large quantities of data. 2013.

R McKerlich, C Ives, and R McGreal. Measuring use and creation of open educational resources in higher education. *International Review of Research in Open and Distance Learning*, 2013.

R Mooney and R Bunescu. Mining knowledge from text using information extraction. *ACM SIGKDD Explorations Newsletter*, 2005.

G Mourad. La segmentation de textes par l'étude de la ponctuation. 1999.

M Nanni and C Rigotti. Extracting trees of quantitative serial episodes. *Knowl-*

edge Discovery in Inductive Databases, 2007.

A Nasr, F Béchet, J Rey, Favre, and J Le Roux. An NLP Tool Suite for Processing Word Lattices. *Proceedings of the ACL-HLT 2011 System Demonstrations*, pages 86–91, 2011.

N Nesselhauf. Corpus Linguistics: A Practical Introduction. 2011.

R. Ng, L Lakshmanan, J Han, and A Pang. Exploratory mining and pruning optimizations of constrained associations rules. *ACM SIGMOD Record*, 1998.

S Nirkhi. Comparative study of Authorship Identification Techniques for Cyber Forensics Analysis. 2013.

M Pecman. L'enjeu de la classification en phraséologie. *Europhras*, pages 127–146, 2004.

J Pedersen and Y Yang. A Comparative Study on Feature Selection in Text Categorization. *Proceeding ICML '97 Proceedings of the Fourteenth International Conference on Machine Learning*, 1997.

J Pei, J Han, and R Mao. CLOSET: An Efficient Algorithm for Mining Frequent Closed Itemsets. *ACM SIGMOD workshop on research issues in data mining and knowledge discovery*, 2000.

J Pei, J Han, B Mortazavi-Asl, H Pinto, Q Chen, U Dayal, and M Hsu. PrefixSpan: mining sequential patterns efficiently by prefix-projected pattern growth. *Proceedings 17th International Conference on Data Engineering*, 2001.

J Pei, J Han, J Wang, H Pinto, and Q Chen. Mining Sequential Patterns by Pattern-Growth : The PrefixSpan Approach. *Ieee Transactions on Knowledge and Data Engineering*, 2004.

M Plantevit. Condensed Representation of Sequential Patterns According to Frequency-Based Measures. 2009.

M Plantevit and T Charnois. Motifs séquentiels pour l'extraction d'information : illustration sur le problème de la détection d'interactions entre gènes. 2009.

S Prasad, V Narsimha, P Reddy, and A Babu. Influence of Lexical, Syntactic and Structural Features and their Combination on Authorship Attribution for Telugu Text. *Procedia Computer Science*, 2015.

S Quiniou, P Cellier, T Charnois, and D Legallois. Fouille de données pour la stylistique : cas des motifs séquentiels émergents. *Actes des Journées Internationales d'Analyse Statistique des Données Textuelles*, 2012.

P Quiniou, Sand Cellier, T Charnois, and D Legallois. What about sequential data mining techniques to identify linguistic patterns for stylistics? *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 7181 LNCS(PART

1):166–177, 2012.

K Rehner, R Mougeon, and T Nadasdi. The Learning of Sociolinguistic Variation By Advanced Fsl Learners . *Studies in Second Language Acquisition*, 2003.

Srikant, R and Agrawal, R. Mining sequential patterns: generalizations and performance improvements. *Proceedings of the 5th International Conference on Extending Database Technology*, 1996.

Y Saeys, Y Saeys, and T Abeel. Robust Feature Selection Using Ensemble Feature Selection Techniques. *European conference on Machine Learning and Knowledge Discovery in Databases (ECML/PKDD)*, 2008.

Sanjuan, E. Ingénierie linguistique et fouille de textes.

T Slimani and A Lazzez. Sequential Mining : Patterns and Algorithms Analysis. 1994.

A Soulet, B Crémilleux, and F Rioult. Condensed representation of emerging patterns. *Pakdd*, pages 127–132, 2004.

A Soulet and B Crémilleux. Adequate condensed representations of patterns. *Data Mining and Knowledge Discovery*, 2008.

R Srikant and R Agrawal. Mining quantitative association rules in large relational tables. *ACM SIGMOD Record*, 1996.

E Stamatatos. A survey of modern authorship attribution methods. *Journal of the American Society for Information Science and Technology*, 2009.

J Swales. Corpus linguistics and English for academic purposes. 1997.

J Toivanen, H Toivonen, A Valitutti, and O Gross. Corpus-Based Generation of Content and Form in Poetry. *Proceedings of the Third International Conference on Computational Creativity*, pages 175–179, 2012.

H Toivonen, M Klemettinen, and P Ronkainen. Pruning and grouping discovered association rules. *ECML'95 Workshop on Statistics, Machine Learning and Knowledge Discovery*, 1995.

Y Toussaint. Fouille de textes : des méthodes' pour la construction d'ontologies et l'annotation sémantique guidée par les connaissances. 2012.

D Tufis, R Ion, and N Ide. Fine-Grained Word Sense Disambiguation Based on Parallel Corpora, Word Alignment, Word Clustering and Aligned Wordnets. page 7, 2005.

N Turenne. Apprentissage statistique pour l'extraction de concepts à partir de textes . Application au filtrage d'informations textuelles . *Sciences-New York*, 2000.

M Verma. Sequential Pattern Mining: A Comparison between GSP, SPADE and Prefix SPAN. 2014.

J Wang and J Han. BIDE: efficient mining of frequent closed sequences. *Proceed-*

ings. 20th International Conference on Data Engineering, 2004.

H Wittmann. Classification linguistique des langues signées non vocalement. *Revue québécoise de linguistique théorique*, 1991.

Y Yang and J Pedersen. A comparative study on feature selection in text categorization. *Machine Learning-International Workshop Then Conference-*, 1997.

X Yan, J Han, and R Afshar. CloSpan: Mining closed sequential patterns in large datasets. *Proc. of SIAM Int. Conf. on Data Mining*, 2003.

M Zaki. SPADE: An efficient algorithm for mining frequent sequences. *Machine Learning*, 2001.

Q Zhao and S Bhowmick. Sequential Pattern Mining : A Survey. *Database*, 2003.